AD-A103 366 SYSTEMS RESEARCH LABS INC DAYTON OHIO F/G 19/5
A GUNNER MODEL FOR TRACER-DIRECTED ANTIAIRCRAFT ARTILLERY FIRE --ETC(U)
JUL 81 K C WEI F33615-79-C-0500
UNCLASSIFIED AFAMRL-TR-81-69 NL

1 OF 1
AD A
103366

END
DATE
FILMED
10-81
DTIC

LEVEL II

(12)

(19) AFAMRL-TR-81-69

AD A103366

(9) Technical rept.,

(6)

A GUNNER MODEL FOR TRACER-DIRECTED ANTIAIRCRAFT ARTILLERY FIRE WITH INTERRUPTED OBSERVATIONS.

(10) KUANG C. WEI
SYSTEMS RESEARCH LABORATORIES, INC.
2800 INDIAN RIPPLE ROAD
DAYTON, OHIO 45440

(11) JUL [redacted]81 (12) 83

(15) F33615-79-C-0500

(16) 6893

(17) 01

AIR FORCE AEROSPACE MEDICAL RESEARCH LABORATORY
AEROSPACE MEDICAL DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

(8)

NOTICES

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from Air Force Aerospace Medical Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22161

Federal Government agencies and their contractors registered with Defense Documentation Center should direct requests for copies of this report to:

Defense Documentation Center
Cameron Station
Alexandria, Virginia 22314

TECHNICAL REVIEW AND APPROVAL

AFAMRL-TR-81-69

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

CHARLES BATES, JR.
Chief
Human Engineering Division
Air Force Aerospace Medical Research Laboratory

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFAMRL-TR-81-69 — 2. GOVT ACCESSION NO.: AD-A103 366	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): A GUNNER MODEL FOR TRACER-DIRECTED ANTI-AIRCRAFT ARTILLERY FIRE WITH INTERRUPTED OBSERVATIONS	5. TYPE OF REPORT & PERIOD COVERED: Technical Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Kuang C. Wei	8. CONTRACT OR GRANT NUMBER(s): F33615-79-C-0500
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Systems Research Laboratories, Inc. 2800 Indian Ripple Road Dayton, Ohio 45440	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62202F/6893/01/35
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Aerospace Medical Research Laboratory, Aerospace Medical Division, Air Force Systems Command, Wright-Patterson AFB, Ohio 45433	12. REPORT DATE: July 1981 13. NUMBER OF PAGES: 83
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

AAA Tracking
AAA Gunner Model
Tracer-Directed Fire
Tracking Error
Observer Theory
Blanking Effect

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In a tracer-directed fire mode, the gunner plays the role of a conventional tracker and a lead angle computer. Tracer rounds are fired continuously by the gunner to provide an additional observation channel. In the event of repeated observation interruption, via blanking the target from the optical display, the gunner's performance is considerably degraded. In this report, a gunner model is designed which consists of a reduced-order observer, a linear feedback controller, and a remnant noise element. The effect of observation interruption is

20. continued

modeled by degrading the model parameters related to the observed states. An exponential decay form is assumed for these parameters. Model parameters and associated time constants are identified from empirical data via a least-squares minimization algorithm. Model predicted tracking and tracer errors are compared with empirical data and found in general agreement with each other. From these results, it is concluded that the gunner model can be used accurately and efficiently in the analysis of the effectiveness of AAA weapon systems under observation blanking.

SUMMARY

This report documents the development of a mathematical model which describes the gunner's performance in an AAA tracer-directed manual firing task under periodic observation interruptions. Observations are interrupted via blanking the target aircraft from the optical display. During the interruption period, the gunner's performance on minimizing tracer-to-target error is considerably degraded. Reduced-order observer theory is applied to design a blanking gunner model. The model consists of a reduced-order observer, a linear feedback controller, and a stochastic remnant element. Both the tracking and the tracer errors are considered measurable. The effect of observation interruption is modeled by degrading the model parameters pertaining to the observed states. An exponential decay form is assumed for these parameters during the blanking and recovery periods. Model parameters and associated time constants are identified systematically from empirical data via a least-squares minimization algorithm. Simulation results for model predictions versus empirical data, over several blanking conditions using a typical helicopter operational trajectory, are included. The results show that the gunner model can adequately describe human response in this compensatory tracking and firing task under observation interruption. The gunner model can be incorporated into existing attrition models to evaluate the survivability of aircraft in tactical engagement scenarios with optical countermeasures present.

PREFACE

This report documents a study performed by Systems Research Laboratories, Inc. (SRL), Dayton, Ohio, for the Air Force Aerospace Medical Research Laboratory (AFAMRL), Human Engineering Division, Optical Countermeasure program. This work was performed under Contract F33615-79-C-0500. The Contract Monitor was Mr. Donald McKechnie and the Program Manager was Maj. Allan M. Dickson. The SRL Project Manager was Mr. Kaile Bishop.

The author extends his appreciation to Dr. Carroll Day, Messrs. Walter Summers and Maris Vikmanis of the Human Engineering Division of the Air Force Aerospace Medical Research Laboratory, Wright-Patterson AFB, for their valuable comments and discussions.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (continued)

Section I

INTRODUCTION

The modeling of human performance in an antiaircraft artillery (AAA) system has been extensively studied by many investigators in the past decade [e.g., Kleinman and Perkins (1974), Phatek et al. (1976), Kou et al. (1978)]. Most of these works dealt with the modeling of human response in a simple tracking task. In the event of interrupted observations, the operator's tracking performance degrades significantly during the interruption period and poses an appealing modeling problem. The author tackled this problem by degrading several observer and controller gains in the model and proved to be rather successful (Yu et al., 1980). Efforts were then directed to study human response in a manual tracking and firing task. In this task, the operator (gunner) directly controls the gun turret and fires tracer rounds continuously toward the target. The gunner perceives the tracer ending position and continuously adjusts weapon pointing in azimuth and elevation to minimize the tracer-to-target error. In this system mode, the gunner has to play both the role of a tracker and a lead angle computer. The conventional tracking task is greatly complicated by the inclusion of lead angle estimation. Wei (1981) developed an observer gunner model which treated the tracer information as delayed measurements. The intent of this paper is to extend the author's previous work to consider an even more general tracking and firing scenario, i.e. to consider a tracking and firing task subject to external measurement interruptions.

The interruptions occur, in the real world, through various electronic/optical countermeasures, weather, or terrain conditions. In this study, extensive manned-simulation experiments were conducted at the Air Force Aerospace Medical Research Laboratory of Wright-Patterson AFB, Ohio. A typical helicopter operational trajectory was used in the experiment. The trajectory consists of three phases. During the first phase, the target is standing still at certain altitude and is half masked by some terrain configuration. At the onset of the second phase, the target pops up for a full unmask flight and moves horizontally. Blanking of target is administered in this phase only. Blanking durations range from 1.5 sec, 3.0 sec, 6.0 sec,

and full blanking. Certain repetition of blanking duration is also included.

The structure of the gunner model in a tracer-directed fire system in Wei (1981) is adopted and generalized here. Nonlinear ballistic equation is used to compute the loci of elevation projectiles. The model consists of a reduced-order observer, a linear feedback controller, and a noise remnant element. The remnant function which lumps all of the random effects due to measurement noise and human neuromotor response noise is assumed to be Gaussian with its covariance being a function of estimated target velocity and acceleration.

The effect of observation interruption is modeled by exponentially degrading the observer gain, the controller gains pertained to observed states, and the bias term in covariance function. Model parameters and time constants are identified separately with respect to no-blanking and blanking empirical data via a least-squares minimization algorithm. The computer simulation of the designed model shows that the model predicted tracking and tracer errors are in good agreement with empirical data over various blanking conditions.

Section II

AAA TRACER-DIRECTED FIRE SYSTEM

In a tracer-directed fire mode, the gunner perceives both the tracking error and the tracer error on a two-dimensional visual display. The tracking error e_1 is the difference between the target angle θ_T and the barrel pointing angle θ_B. It is also referred to as "lag angle" later in this report. The tracer error e_2 is the difference between the target angle θ_T and the projectile ending angle θ_p. In the simulation experiment, the projectile flight path ended at the range of the target. Each tracer round disappeared at this point, θ_p, from the display. The detailed description of the configuration is described in Wei (1981). We will briefly summarize the underlying dynamic system in this report. Figure 1 is the block diagram of an AAA tracer-directed fire system.

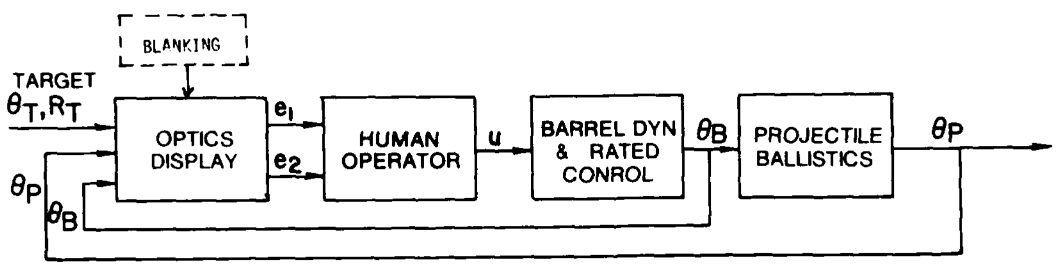


Figure 1. Block Diagram of an AAA Tracer-Directed Fire System

At any given time, the target trajectory input θ_T is fed into a visual display device and combined with the barrel pointing angle θ_B, as well as the projectile ending angle θ_p, to form error signals e_1 and e_2. The human operator observes these error signals and generates a control output u via a controller, or H-grip, displacement. The control signal then drives the barrel and rate control plant for a new barrel pointing angle θ_B. Tracer round is fired at this angle and passes through the projectile ballistics computation to obtain the projectile ending angle θ_p. The task of the gunner is to constantly align the projectile ending angle to the target angle, i.e. to minimize the tracer error e_2. The dynamics for the elevation and the azimuth firing system are very similar. In addition, the elevation

system can be decoupled from the azimuth system. However, the azimuth system cannot be separated from the elevation system due to a coupling factor cos $(\theta_B)_{EL}$ in the measurement equation.

By introducing a state vector $\underline{x}_i(t) = [x_{i1}(t), x_{i2}(t), x_{i3}(t)]^T$, "T" means "the transpose of," with $x_{i1}(t) \triangleq \theta_{iT}(t) - \theta_{iB}(t)$, $x_{i2}(t) \triangleq \theta_{iT}(t) - \theta_{iP}(t)$ and $x_{i3} = \dot{\theta}_{iT}(t)$, i = 1, 2*, the following system and measurement equations which represent the underlying tracer-directed fire system can be derived, see Wei (1981).

$$\dot{\underline{x}}_i = \underline{A}_i\underline{x}_i + \underline{B}_i\, u_i(t) + \underline{E}_i(t)\, u_i(t-\tau) + \underline{F}_i\, \ddot{\theta}_{iT}(t) + \underline{G}_i(t) \qquad (1)$$

and

$$\underline{y}_i(t) = \underline{C}_i(t)\, \underline{x}_i(t) \qquad i = 1,2 \qquad (2)$$

where

$$\underline{A}_i = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \qquad \underline{B}_i = \begin{bmatrix} b_i \\ 0 \\ 0 \end{bmatrix} \qquad \underline{E}_i(t) = \begin{bmatrix} 0 \\ e_i(t) \\ 0 \end{bmatrix}$$

$$\underline{F}_i = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \underline{G}_i(t) = \begin{bmatrix} 0 \\ g_i(t) \\ 0 \end{bmatrix} \qquad \underline{C}_i(t) = \begin{bmatrix} c_i & 0 & 0 \\ 0 & c_i & 0 \end{bmatrix}$$

---

*If not otherwise specified, the first subscript index i represents the elevation (i = 1) or azimuth axis (i = 2), while the second index represents the i-th element or row of a matrix.

with

$$b_1 = -1.34$$

$$b_2 = -1.28$$

$$c_1 = 1$$

$$c_2 = \cos \theta_{1B}(t)$$

$$e_1(t) = -1.34 \times (1-\dot{\tau}) \times \left[1 + (0.0052\tau + 0.000486\tau^2) \sin \theta_{1B}(t-\tau)\right]$$

$$e_2(t) = -1.28 \times (1-\dot{\tau})$$

$$g_1(t) = (0.0052+0.000972\tau) \times \tau \times \cos \theta_{1B}(t-\tau)$$

$$g_2(t) = 0$$

$\ddot{\theta}_{iT}$, u_i, y_{i1}, and y_{i2} denote the elevation or azimuth components of the target acceleration, the gunner's control output and the observed tracking error (lag angle) and tracer error, respectively.

If we introduce a transformation on the states x_{i1} and x_{i2} by $x'_{i1} = c_i x_{i1}$, $x'_{i2} = c_i\, x_{i2}$, $x'_{i3} = x_{i3}$ then Equations (1)-(2) can be rewritten as follows.

$$\dot{\underline{x}}'_i = \underline{A}'_i(t)\, \underline{x}'_i + \underline{B}'_i(t)\, u_i(t) + \underline{E}'_i(t)\, u_i\,(t-\tau)$$

$$+ \underline{F}_i\, \ddot{\theta}_{iT}(t) + \underline{G}'_i(t) \tag{3}$$

$$\underline{y}_i(t) = \underline{C}'_i\, \underline{x}'_i(t) \tag{4}$$

where

$$\underline{A}_i'(t) = \begin{bmatrix} \dot{c}_i c_i^{-1} & 0 & c_i \\ 0 & \dot{c}_i c_i^{-1} & c_i \\ 0 & 0 & 0 \end{bmatrix} \qquad \underline{B}_i'(t) = \begin{bmatrix} c_i b_i \\ 0 \\ 0 \end{bmatrix}$$

$$\underline{E}_i'(t) = \begin{bmatrix} 0 \\ c_i e_i(t) \\ 0 \end{bmatrix} \qquad \underline{G}_i'(t) = \begin{bmatrix} 0 \\ c_i g_i(t) \\ 0 \end{bmatrix} \qquad \underline{F}_i = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\underline{C}_i' = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \qquad i = 1,2$$

and $\underline{x}_i'(t) = \left[x_{i1}'(t),\ x_{i2}'(t),\ x_{i3}'(t)\right]^T$. Notice that the coupling factor $\cos\theta_{1B}$ is removed from the measurement equation and absorbed into the system equation. Equations (3)-(4) represent a nonhomogeneous linear time-varying system with a time-varying delay in the control.

Section III

AN AAA GUNNER BLANKING MODEL

In Wei (1981), the author proposed an observer gunner model for gunner performance in a tracer-directed fire system. The function of the gunner can be decomposed into two parts to be modeled. In the first part, the gunner observes continuous signals and makes an estimate of system states based on his internal model of target motion. In the second part, the gunner utilizes the observed and estimated states to form and exercise a control action in order to achieve his objective. The former one corresponds to an estimation process, while the latter corresponds to a control process. The reduced-order observer is used in conjunction with a linear feedback control law to model the gunner's function. The structure of the model is shown in Figure 2.

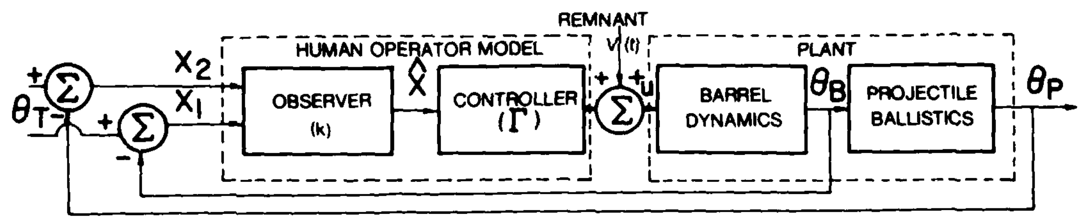


Figure 2. Block Diagram of an AAA Gunner Model

This model structure is retained for the blanking case except the ballistic equation is no longer parameterized by a linear equation relating θ_P and θ_B. Instead, a more realistic nonlinear ballistic equation is used as shown in Equations (3)-(4). In addition, time-varying gains are used to model the effect of observation interruption. We will discuss the no-blanking case first, then the blanking case.

NO BLANKING

The equation representing the gunner's <u>internal</u> model of the tracking and firing system can be written as

$$\dot{\underline{x}}_i'(t) = \underline{A}_i'(t)\underline{x}_i'(t) + \underline{B}_i'(t)u_i(t) + \underline{E}_i'(t)\ u_i(t-\tau) + \underline{G}_i'(t) \tag{5}$$

$$\underline{y}_i(t) = \underline{C}_i'\ \underline{x}_i'(t) \qquad i = 1,2 \tag{6}$$

Since both x_{i1}' and x_{i2}' are measurable, the only state that needs to be estimated is x_{i3}. The state reconstructor equation for x_{i3} can be derived by applying the reduced-order observer theory (Luenberger, 1971)

$$\begin{aligned}\dot{\hat{x}}_{i3}(t) = &-(k_{i1}+k_{i2})c_i\hat{x}_{i3}(t) + k_{i1}\dot{y}_{i1}(t) + k_{i2}\dot{y}_{i2}(t)\\ &-b_ik_{i1}c_iu_i(t) - b_ik_{i2}c_ie_i(t)u_i(t-\tau) - k_{i2}c_ig_i(t)\\ &- k_{i1}\dot{c}_ic_i^{-1}y_{i1}(t) - k_{i1}\dot{c}_ic_i^{-1}y_{i2}(t)\end{aligned} \tag{7}$$

The objective of the gunner is to minimize the tracer error so that a maximum probability of hit could result. In other words, the gunner's response in the control process would be to stabilize the underlying system, especially the tracer error $x_{i2}'(t)$; therefore, a linear feedback control law of the following form is designed to achieve this objective.

$$u_i(t) = \underline{\Gamma}_i\ \hat{\underline{x}}_i'\ (t) + v_i(t) \tag{8}$$

where

$$\underline{\Gamma}_i = [\gamma_{i1}, \gamma_{i2}, \gamma_{i3}]$$

is a vector of controller gains to be identified,

$$\underline{\hat{x}}_i'(t) = [y_{i1}(t), y_{i2}(t), \hat{x}_{i3}(t)]^T$$

is a vector of measurable states and estimated state, $v_i(t)$ is a remnant noise function assumed to be Gaussian with zero mean and a covariance function

$$E\left[v_i(t)\, v_i(s)\right] = \left[\alpha_{i1} + \alpha_{i2}\left|\hat{\dot{\theta}}_{iT}(t)\right| + \alpha_{i3}\left|\hat{\ddot{\theta}}_{iT}(t)\right|\right]\delta(t-s) \tag{9}$$

for all t and s. α_{ij} are nonnegative model parameters to be determined. $\hat{\dot{\theta}}_{iT}$ and $\hat{\ddot{\theta}}_{iT}$ are estimated target angle rate and acceleration, respectively. Equations (7) and (8) represent the gunner's response in the estimation and control process of the tracking and firing task. If we define a new state vector

$$\underline{X}_i(t) = \left[y_{i1}(t), y_{i2}(t), x_{i3}(t), x_{i3}(t) - \hat{x}_{i3}(t)\right]^T$$

then the state equation of the closed-loop system is obtained by combining Equations (7) and (8) with Equations (3) and (4) of the <u>actual</u> tracking and firing system.

$$\begin{aligned}\dot{\underline{X}}_i(t) = {} & \underline{A}_i(t)\underline{X}_i(t) + \underline{D}_i(t)\,\underline{X}_i(t-\tau) + \underline{F}_i\ddot{\theta}_{iT}(t) + \underline{E}_{i0}(t)v_i(t) \\ & + \underline{E}_{i1}(t)v_i(t-\tau) + \underline{R}_i(t)\end{aligned} \tag{10}$$

where

$$
\underline{A}_i(t) = \begin{bmatrix} \dot{c}_i c_i^{-1} + b_i c_i \gamma_{i1} & b_i c_i \gamma_{i2} & (1+b_i\gamma_{i3})c_i & -b_i c_i \gamma_{i3} \\ 0 & \dot{c}_i c_i^{-1} & c_i & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -k_i c_i \end{bmatrix}
$$

$$
\underline{D}_i(t) = c_i e_i(t) \quad \cdot \quad \begin{bmatrix} 0 & 0 & 0 & 0 \\ \gamma_{i1} & \gamma_{i2} & \gamma_{i3} & -\gamma_{i3} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}
$$

$$
\underline{F}_i = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \qquad \underline{E}_{i0}(t) = \begin{bmatrix} b_i c_i \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \underline{E}_{i1}(t) = \begin{bmatrix} 0 \\ c_i e_i(t) \\ 0 \\ 0 \end{bmatrix} \qquad \underline{R}_i(t) = \begin{bmatrix} 0 \\ c_i g_i(t) \\ 0 \\ 0 \end{bmatrix}
$$

$$
k_i \triangleq k_{i1} + k_{i2} \qquad\qquad i = 1, 2
$$

There are seven model parameters in total, i.e., k_i, γ_{i1}, γ_{i2}, γ_{i3} α_{i1}, α_{i2}, and α_{i3} that need to be determined from the empirical no-blanking tracking data.

BLANKING

The optical display of target was blanked periodically according to the duty cycles and durations listed in Table 1.

TABLE 1. BLANKING CONDITIONS

Condition	Duty Cycle (%)	Blanking duration (sec)
1	25	1.5
2	25	3.0
3	25	6.0
4	50	1.5
5	50	3.0
6	50	6.0
7	75	1.5
8	75	3.0
9	75	6.0
10	100	1.5

The duty cycle is defined as the ratio of the blanking duration to the cycle time. The blanking duration is the length of time that the target is blanked so that the subject cannot see the target. The blanking always occurs at the last portion of a cycle and may reoccur periodically over the entire TOW firing period. An example of blanking sequence is given in Figure 3.

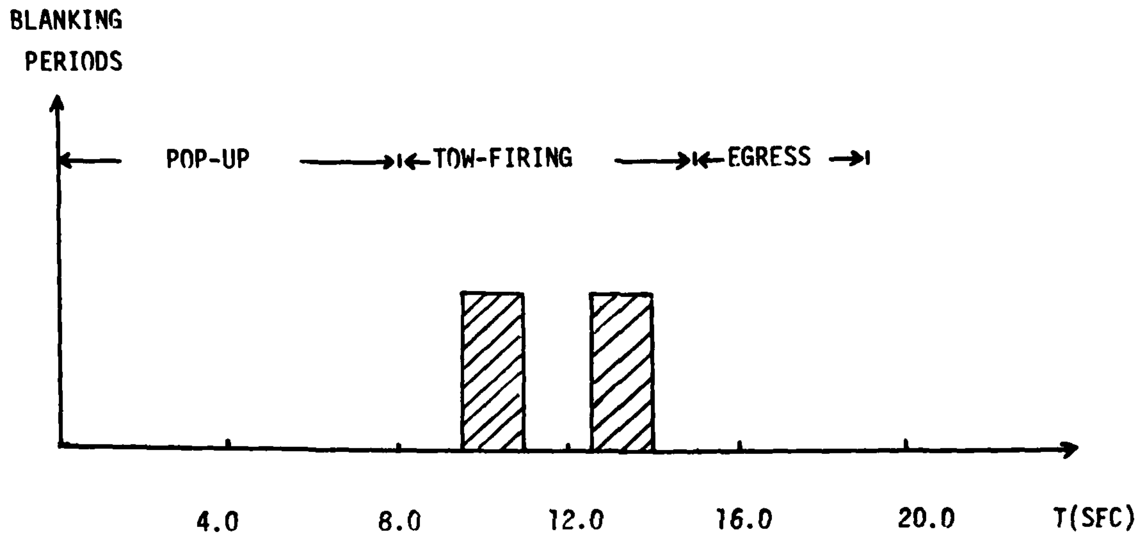


Figure 3. Sequence of Blanking for Condition 4

The gunner's performance deteriorates considerably under observation interruption via blanking the target. In Yu (1981), the effect of blanking on the gunner's tracking performance was modeled successfully by degrading the gunner's estimation gain k(t) and the controller gain $\gamma(t)$. A similar approach is adopted here to model the effect of blanking in a more complex firing task. More specifically, the observer gain k_i and controller gains γ_{i1} and γ_{i2} which pertain to the observed states x'_{i1} and x'_{i2} are assumed to decrease exponentially as the blanking starts and to increase exponentially as the blanking stops (see Figure 4).

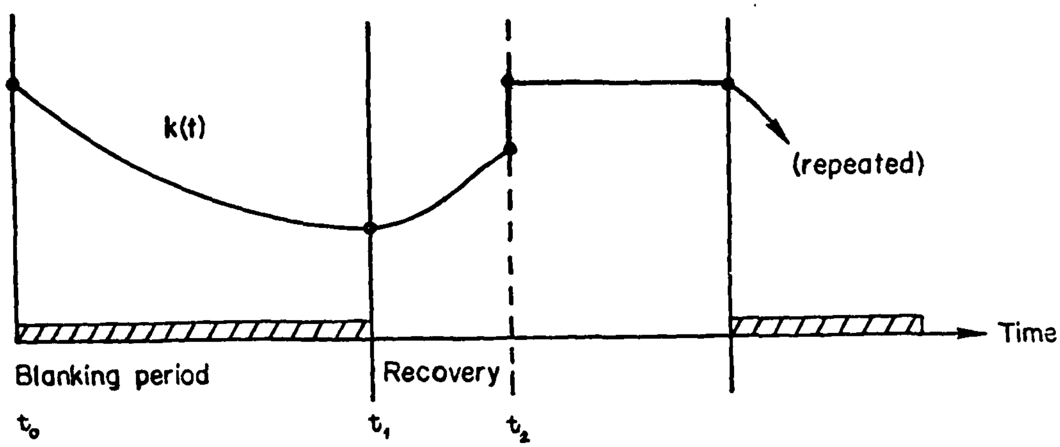


Figure 4. Degradation of Model Parameters in Blanking and Recovery Period

Given a blanking period $[t_0, t_1]$ followed by a recovery period $[t_1, t_2]$, the degradation of gains can be expressed by the following equations.

During the blanking period:

$$k_i(t) = k_i(t_0) \exp\left(-\frac{t-t_0}{\tau_{ik}}\right) \tag{11}$$

$$\gamma_{i1}(t) = \gamma_{i1}(t_0) \exp\left(-\frac{t-t_0}{\tau_{i\gamma_1}}\right) \tag{12}$$

$$\gamma_{i2}(t) = \gamma_{i2}(t_0) \exp\left(-\frac{t-t_0}{\tau_{i\gamma_2}}\right) \tag{13}$$

$$\alpha_{i1}(t) = \alpha_{i1}(t_0) \left[1 - \exp\left(-\frac{t-t_0}{\tau_{i\alpha_1}}\right)\right] \tag{14}$$

During the recovery period:*

$$k_i(t) = k_i(t_1) + \left[k_i(t_0) - k_i(t_1)\right] \left[1 - \exp\left(-\frac{t-t_1}{\tau_{ik}}\right)\right] \tag{15}$$

$$\alpha_{i1}(t) = \alpha_{i1}(t_1) \left[1 - \exp\left(-\frac{t-t_1}{\tau_{i\alpha_1}}\right)\right] \tag{16}$$

The time constants τ_{ij} associated with each gain parameter, $\alpha_{i1}(t_0)$, and $\alpha_{i1}(t_1)$ are determined from the empirical tracking data collected in the blanking experiments, as shown in the next section.

---

*In the simulation program, the length of the recovery period is defined as the minimum of 1.5 sec and one-third of blanking period to avoid covariance being negative.

Section IV
PARAMETER IDENTIFICATION AND SIMULATION

The least-squares identification program developed in Wei (1981) was modified to identify the no-blanking parameters. Equation (10) can be first decoupled and then approximated, via the Average Approximation Method, by the following ordinary differential equation (Banks and Burns, 1978).

$$\dot{\underline{W}}_i(t) = \underline{N}_i(t)\ \underline{W}_i(t) + \underline{M}_i\ \underline{n}_i(t) \tag{17}$$

$$\dot{x}_{i3}(t) = \ddot{\theta}_{iT}(t) \tag{18}$$

$$\dot{x}_{i4}(t) = -k_i(t)c_i\ x_{i4}(t) + \ddot{\theta}_{iT}(t) \tag{19}$$

where

$$\underline{N}_i = \begin{bmatrix} \dot{c}_i c_i^{-1} + b_i c_i \gamma_{i1}(t) & b_i c_i \gamma_{i2}(t) & 0 & 0 \\ 0 & \dot{c}_i c_i^{-1} & c_i \gamma_{i1}(t) e_i & c_i \gamma_{i2}(t) e_i \\ \frac{1}{\tau} & 0 & -\frac{1}{\tau} & 0 \\ 0 & \frac{1}{\tau} & 0 & -\frac{1}{\tau} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad \underline{W}_i = \begin{bmatrix} x'_{i1}(t) \\ x'_{i2}(t) \\ x'_{i1}(t-\tau) \\ x'_{i2}(t-\tau) \end{bmatrix}$$

$$\underline{n}_i = \begin{bmatrix} \left(1 + b_i\gamma_{i3}\right) c_i \; x_{i3}(t) - b_i c_i \gamma_{i3} x_{i4}(t) + b_i c_i v_i(t) \\ c_i x_{i3}(t) + c_i e_i(t) \left\{\gamma_{i3} x_{i3}(t-\tau) - \gamma_{i3} x_{i4}(t-\tau) + v_i(t-\tau)\right\} + c_i g_i(t) \end{bmatrix}$$

IDENTIFICATION OF MODEL PARAMETERS

The equation which governs the mean of states is obtained by taking expectation of Equation (17):

$$\dot{\overline{\underline{W}}}_i(t) = \underline{N}_i(t)\, \overline{\underline{W}}_i(t) + \underline{M}_i\, \underline{\xi}_i(t) \tag{20}$$

where

$$\overline{\underline{W}}_i(t) = \left[E\left\{x'_{i1}(t)\right\},\ E\left\{x'_{i2}(t)\right\},\ E\left\{x'_{i1}(t-\tau)\right\},\ E\left\{x'_{i2}\ (t-\tau)\right\}\right]^T$$

$$\underline{\xi}_i(t) = \begin{bmatrix} (1+b_i\gamma_{i3})c_i x_{i3}(t) - b_i c_i \gamma_{i3} x_{i4} \\ \\ c_i x_{i3}(t) + c_i e_i(t)\gamma_{i3}\left\{x_{i3}(t-\tau) - x_{i4}(t-\tau)\right\} + c_i g_i(t) \end{bmatrix}$$

The first and second component of $\overline{\underline{W}}_i$ represent the model prediction of ensembled mean of tracking and tracer error, respectively. On the other hand, the covariance matrix $\underline{P}_i(t)$ satisfies the following equation:

$$\dot{\underline{P}}_i(t) = \underline{N}_i(t)\, \underline{P}_i(t) + \underline{P}_i(t)\, \underline{N}_i^T(t) + \underline{L}_i(t)\, \underline{Q}_i(t)\, \underline{L}_i^T(t) \tag{21}$$

where

$$\underline{P}_i(t) = E\left\{\left[\underline{W}_i(t) - \overline{\underline{W}}_i(t)\right]\left[\underline{W}_i(t) - \overline{\underline{W}}_i(t)\right]^T\right\}$$

$$
\underline{L}_i(t) = \begin{bmatrix} b_i c_i & 0 \\ 0 & c_i e_i(t) \\ 0 & 0 \\ 0 & 0 \end{bmatrix}
$$

$$
\underline{Q}_i(t) = \begin{bmatrix} \alpha_{i1}(t) + \alpha_{i2}\left|\hat{\ddot{\theta}}_{iT}(t)\right| + \alpha_{i3}\left|\hat{\ddot{\theta}}_{iT}(t)\right| & 0 \\ 0 & \alpha_{i1}(t-\tau) + \alpha_{i2}\left|\hat{\dot{\theta}}_{iT}(t-\tau)\right| + \alpha_{i3}\left|\hat{\ddot{\theta}}_{iT}(t-\tau)\right| \end{bmatrix}
$$

The first and second diagonal element $p_{i11}(t)$ and $p_{i22}(t)$ of $\underline{P}_i(t)$ represent the square of the model prediction of standard deviation of tracking and tracer error, respectively. Notice that time-varying parameters are assumed for k_i, γ_{i1}, γ_{i2}, and α_{i1} to reflect the effect of blanking. Since the blanking effect to the estimation of target velocity $x_{i3}(t)$ is predominantly expressed through degradation of k_i, there is no need to consider a time-varying γ_{i3}, α_{i2}, and α_{i3}.

The steady-state value of the parameters are first identified via a least-squares curve-fitting identification program. The reference curves to be fitted are obtained from empirical tracking and tracer data collected in the manned simulation experiments without observation interruption. These experiments were conducted on an AAA simulator at the Air Force Aerospace Medical Research Laboratory. Three simulated helicopter trajectories ranged 1500 M, 2000 M, and 2500 M from the AAA system were used as target trajectories. Figure 5 shows some characteristics for the 1500 M trajectory. Let $\bar{x}_{i1}(t)$ and $\bar{x}_{i2}(t)$ be the empirical ensemble means of tracking and tracer errors and $\bar{s}_{i1}(t)$ and $\bar{s}_{i2}(t)$ be the corresponding standard deviations. These empirical means and standard deviations were obtained by averaging and computing the variance of the empirical data from 40 simulation runs with the same target trajectory and the same subject.

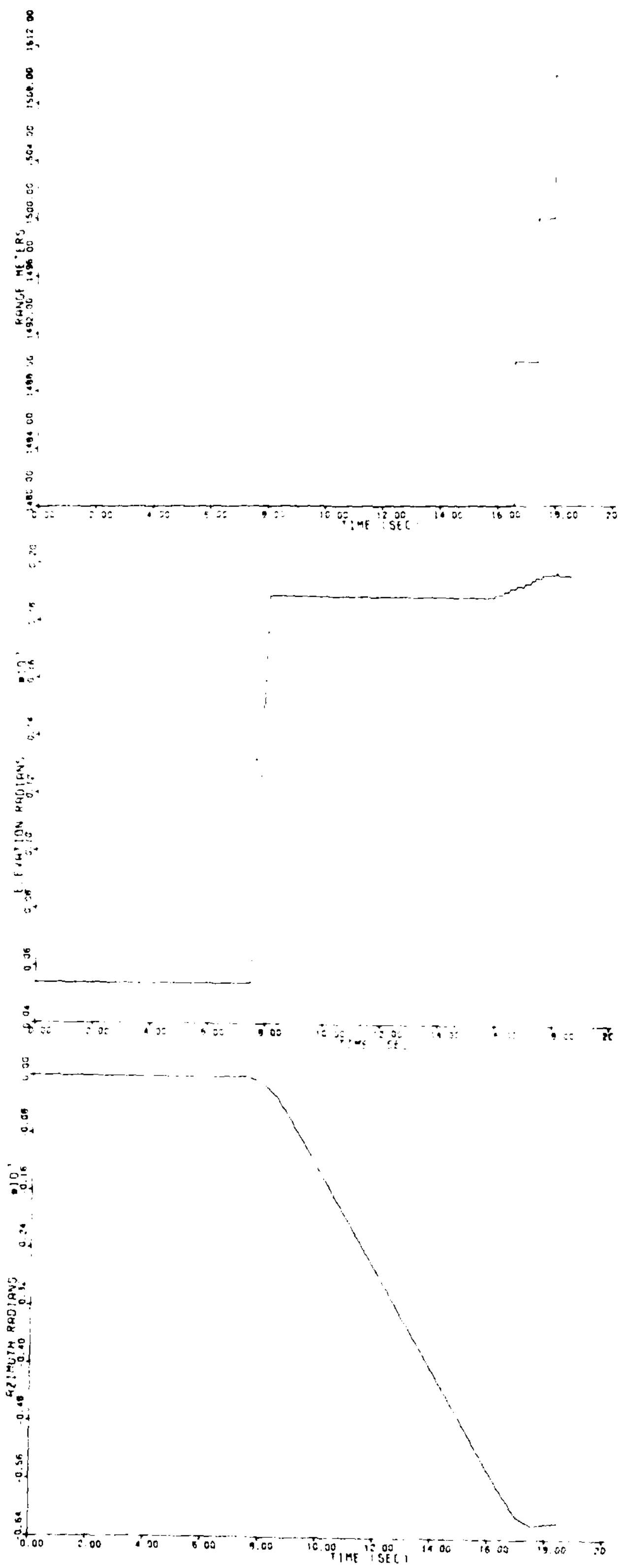


Figure 5. Trajectory Characteristics

The parameters were identified by minimizing the cost function

$$J_i\left[k(t_s), \underline{\Gamma}(t_s), \underline{\alpha}(t_s)\right]$$

defined as follows:

$$\min_{k,\underline{\Gamma},\underline{\alpha}} J_i\left[k(t_s), \underline{\Gamma}(t_s), \underline{\alpha}(t_s)\right] = \min_{k,\underline{\Gamma},\underline{\alpha}} \sum_{j=1}^{2} \int_{t_s}^{t_f} \left\{\left[c_i^{-1} w_{ij}(t) - \bar{x}_{ij}(t)\right]^2 + \ell_i \left[c_i^{-1} p_{ijj}^{\frac{1}{2}}(t) - \bar{s}_{ij}(t)\right]^2\right\} dt \qquad (22)$$

$$i = 1,2$$

where t_s is the initial time when a selected tracer round reaches the range of the target, t_f is the time when the last tracer round is fired, ℓ_i is a positive weighting factor chosen to be one in the identification runs.

The direct search algorithm developed in Wei (1981) was modified to identify the steady-state values of the parameters. The tracking and tracer data of the helicopter trajectory, ranged 1500 M from the AAA simulator, without blanking, were used to obtain the following steady-state parameter values shown in Table 2.

The time constants associated with the parameters were determined empirically from the data of the 1500 M trajectory with blanking condition 5 and listed in Table 3.

TABLE 2. STEADY-STATE PARAMETER VALUES

Parameter / Gunner Model	Observer Gain	Controller Gains			Coefficients of Covariance Function		
	$K(t_0)$	$\gamma_1(t_0)$	$\gamma_2(t_0)$	γ_3	$\alpha_1(t_0)$	α_2	α_3
Elevation	1.5471	0.017491	0.024433	0.42318	0.22446E-7	0.17975E-3	0.173-ZE-3
Azimuth	6.5394	0.13894	0.17773	1.0353	0.25286E-5	0.19766E-3	0.75785E-3

TABLE 3. TIME CONSTANTS ASSOCIATED WITH PARAMETER VECTOR

Time Constants / Gunner Model	Blanking Period					Recovery Period		
	τ_k	τ_{γ_1}	τ_{γ_2}	τ_{α_1}	$\alpha_1(t_0)$	τ_k	τ_{α_1}	$\alpha_1(t_1)$
Elevation	13.23	*	1.92	8.33	0.0001	2.33	2.33	-0.0001
Azimuth	13.23	8.33	1.92	8.33	0.001	2.33	2.33	-0.001

*Insensitive for elevation case, no degradation is necessary.

Notice that the time constants for τ_k, τ_{γ_2}, and τ_{α_1} are the same for both elevation and azimuth gunner model. This is as expected because the gunner manipulates the H-grip for elevation and azimuth tracking indiscriminantly with respect to the observation interruption. On the other hand, $\alpha_1(t_0)$ and $\alpha_1(t_1)$ for the azimuth case are considerably greater than that for the elevation case. This reflects the steeper increase of uncertainty to the target's position along the azimuth axis, because the azimuth component of target acceleration is much higher than the elevation component.

SIMULATION RESULTS

The gunner model was implemented on a CDC CYBER 175 computer to simulate the man-in-the-loop AAA tracking and firing task. For the convenience of numerical computation, Equations (20) and (21) are discretized into the following form:

$$\underline{\bar{W}}_i^{\,n+1} = \underline{\Phi}_i^n \, \underline{\bar{W}}_i^n + \underline{H}_i^n \, \underline{\xi}_i^n \tag{23}$$

$$\underline{P}_i^{n+1} = \underline{\Phi}_i^n \, \underline{P}_i^n \left(\underline{\Phi}_i^n\right)^T + \frac{1}{\Delta} \, \underline{R}_i^n \, \underline{Q}_i^n \left(\underline{R}_i^n\right)^T \tag{24}$$

where

$$t_{n+1} = t_0 + (n+1)\Delta$$

$$\underline{\bar{W}}_i^{n+1} = \underline{W}_i(t_{n+1})$$

$$\underline{P}_i^{n+1} = \underline{P}_i(t_{n+1})$$

$$\underline{\Phi}_i^n = \exp\,[\underline{N}_i(t_n)\ \Delta]$$

$$\underline{H}_i^n = \int_0^\Delta \exp[\underline{N}_i(t_n) \cdot \sigma]\, d\sigma \cdot \underline{M}_i$$

$$\underline{R}_i^n = \int_0^\Delta \exp[\underline{N}_i(t_n) \cdot \sigma]\, d\sigma \cdot \underline{L}_i(t_n)$$

$$\underline{\xi}_i^n = \underline{\xi}_i(t_n)$$

$$\underline{Q}_i^n = \underline{Q}_i(t_n)$$

$$\Delta = 0.06 \text{ seconds}$$

A simulation program was developed which uses the recursive Equations (23) and (24) to simulate a closed-loop AAA tracking and firing task. Inputs to the simulation program are the time history of range and acceleration of the target aircraft, the initial angular position and velocity of the target, the number of blanking intervals, and the blanking intervals in chronological order. Outputs of the simulation program are model predicted mean tracking error and its standard deviation.

Simulation results are shown in Figures 6 through 17 for the blanking conditions 3, 4, 5, 6, 7, and 10. The solid curves in these figures are the empirical data which are obtained by averaging the results of 40 experimental runs. The dashed curve is the model prediction of ensembled mean and standard deviation.

Figure 6 and Figure 7 show the comparison of model versus empirical elevation mean and standard deviation, azimuth mean and standard deviation of both tracking errors (lag), and tracer errors for the no-blanking

case. Figure 10 and Figure 11 show the results for blanking condition 5 which had a 50 percent, 3.0 second blanking occur during [11.01, 14.01] seconds.

Of particular interest is the comparison of the empirical standard deviation in Figure 6 and Figure 10. The effect of blanking the target to gunner's performance is clearly demonstrated by the sharp increase of the standard deviation of tracking errors during the blanking period [11.01, 14.01] seconds. This effect is very well modeled by degrading selected model parameters as indicated in the model prediction curve in Figure 8. Similar agreements between the empirical data and the model prediction can be found in Figures 8, 9, and 12 through 17.

These figures show that the designed gunner model can provide consistent prediction of the gunner's empirical tracking data as well as the tracer error data for both no-blanking and blanking cases. These figures also demonstrate that, for a given AAA weapon system, the same set of parameter values and associated time constants can be used to predict the human tracking and tracer errors for all simulated blanking conditions.

However, due to the fact that the helicopter trajectory has very low elevation axis maneuvering, these parameters may only hold for similar types of low maneuvering helicopter trajectories. Reidentification of these parameters may be needed for other high maneuvering trajectories. The computer execution time of the overall simulation for an 18 second helicopter trajectory takes about 5.60 cp seconds on a CDC CYBER 175 computer.

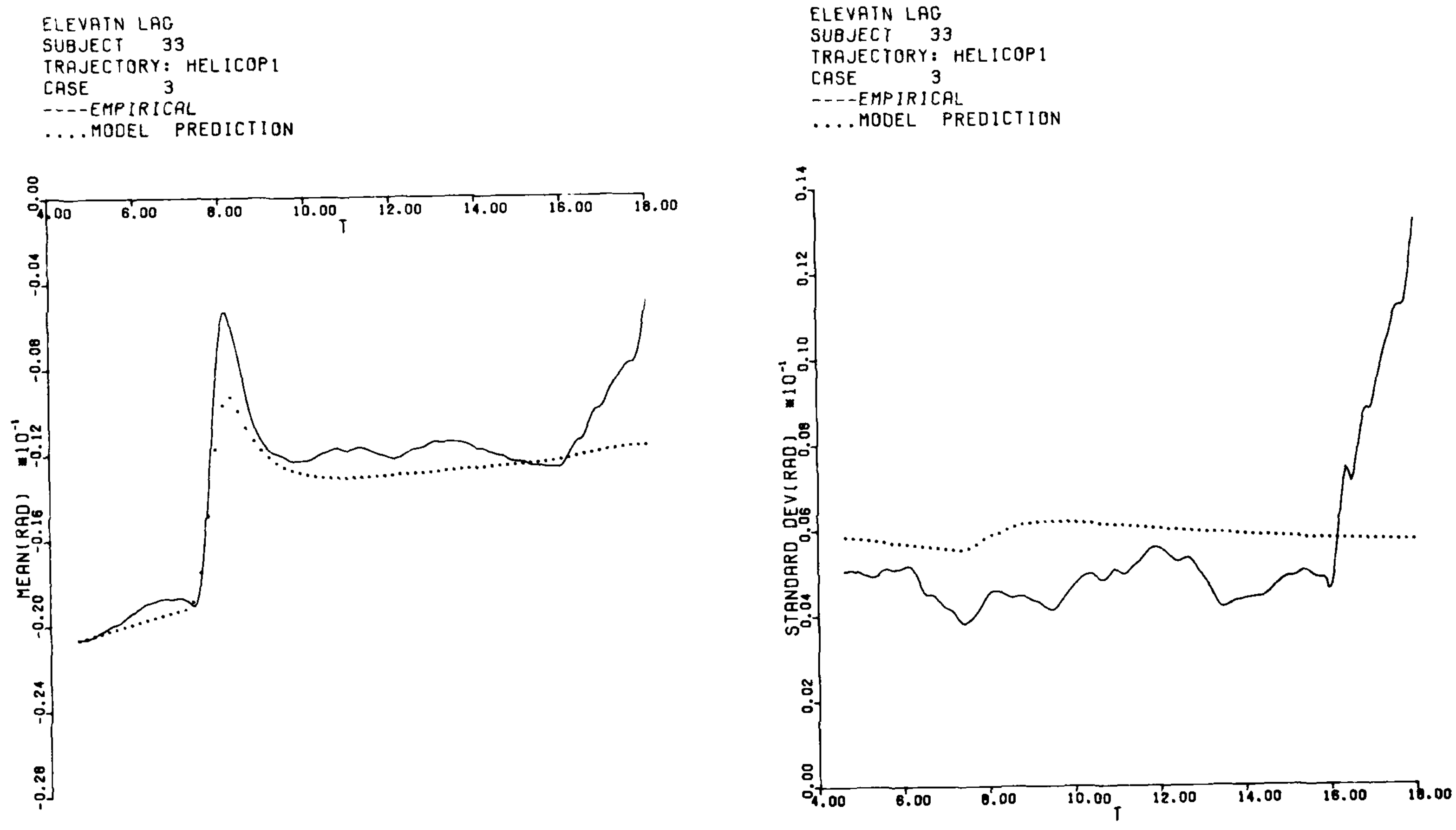


Figure 6a. Mean and Standard Deviation of Tracking Error—Elevation—No Blanking

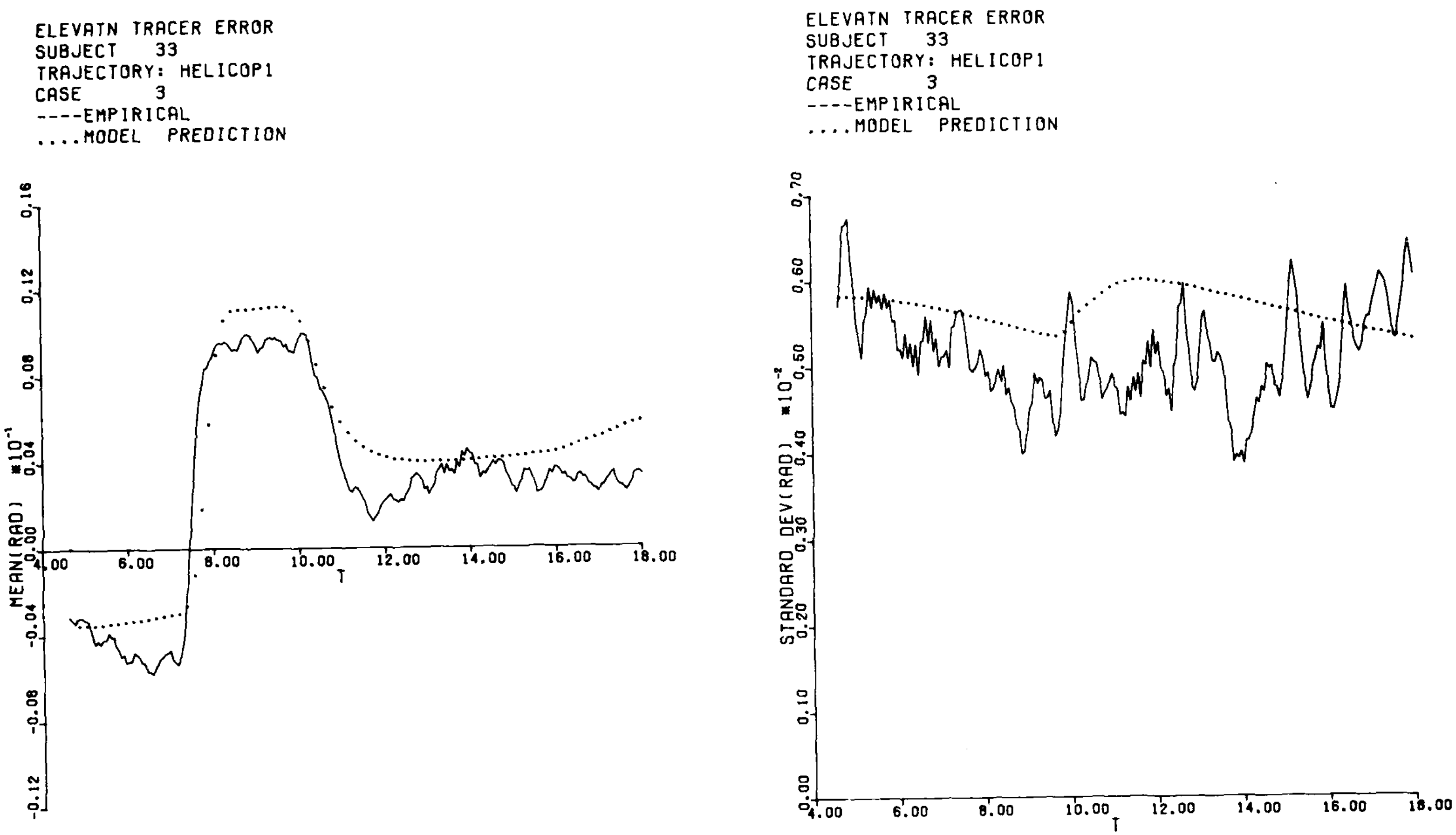


Figure 6b. Mean and Standard Deviation of Tracer Error--Elevation--No Blanking

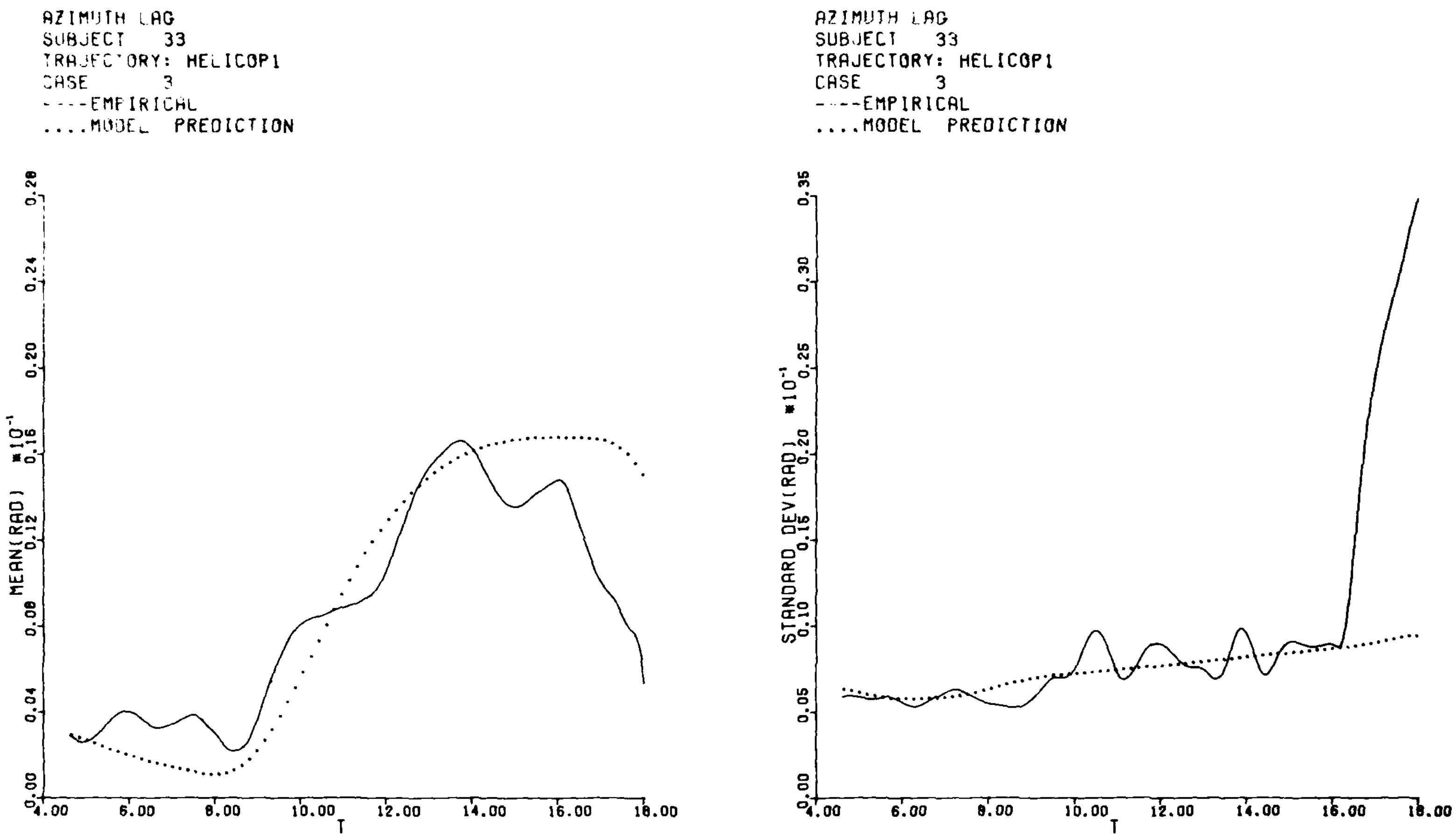


Figure 7a. Mean and Standard Deviation of Tracking Error--
Azimuth--No Blanking

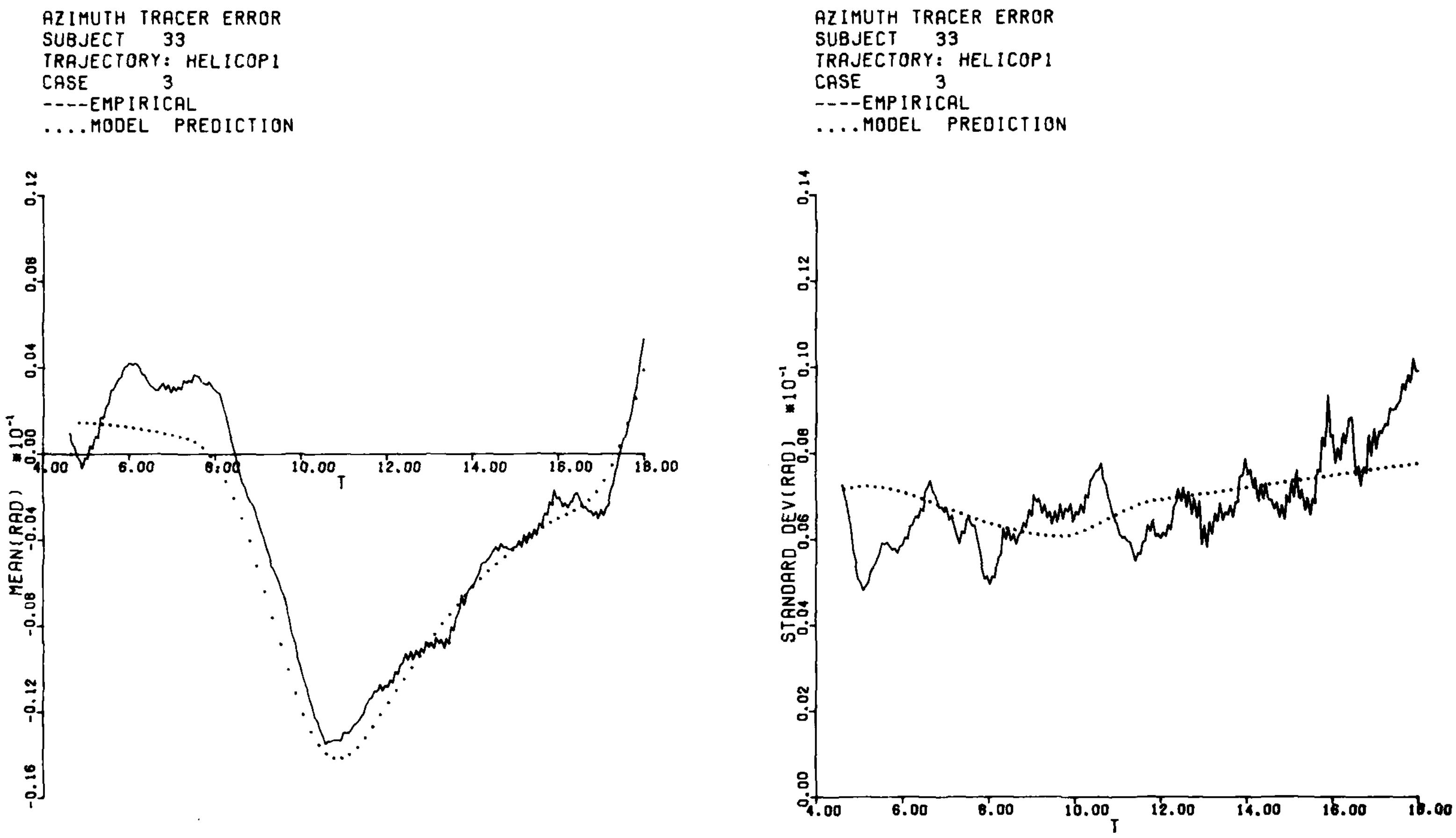


Figure 7b. Mean and Standard Deviation of Tracer Error--Azimuth--No Blanking

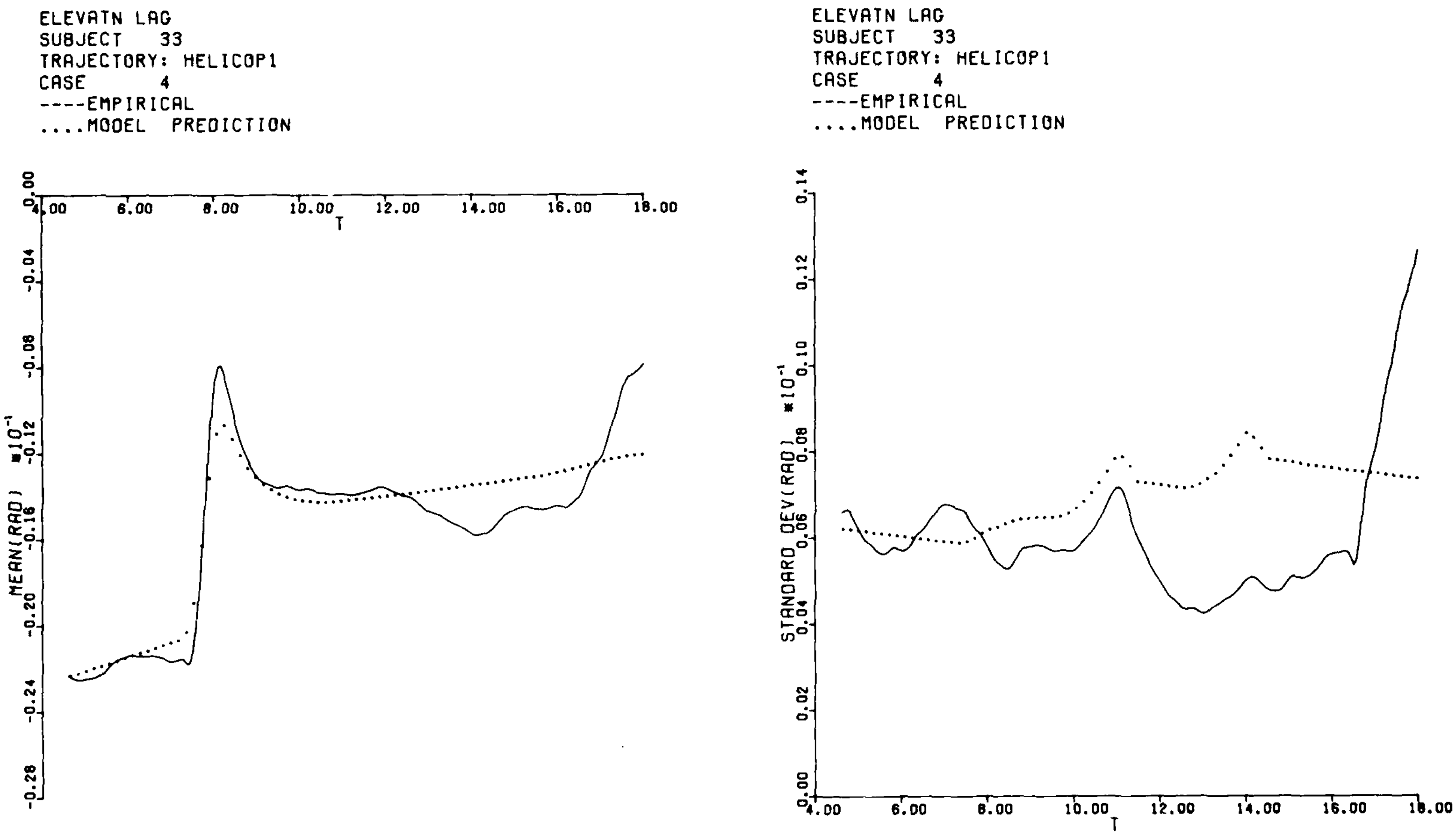


Figure 8a. Mean and Standard Deviation of Tracking Error—Elevation—1.5 Seconds, 50 Percent Blanking

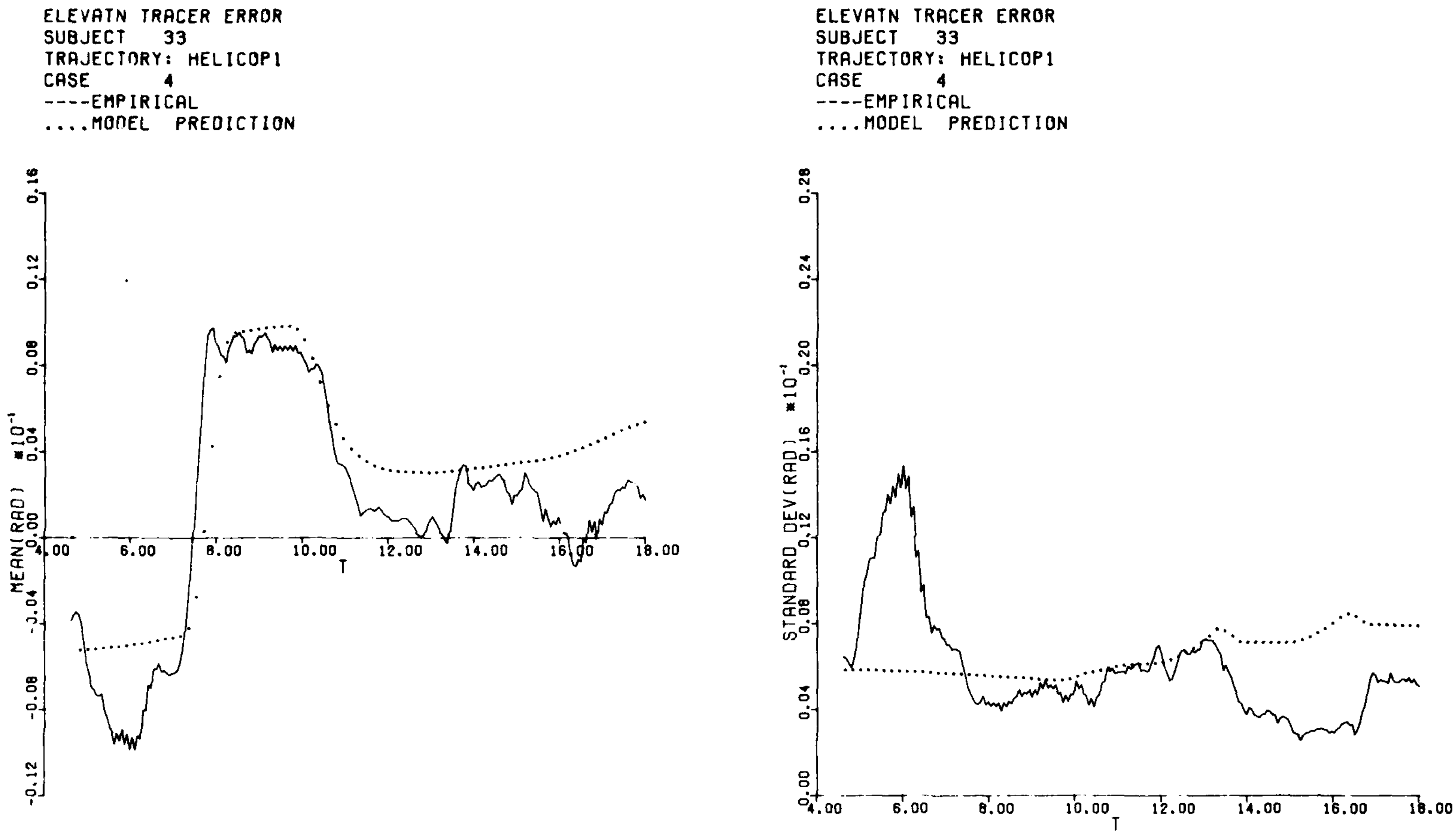


Figure 8b. Mean and Standard Deviation of Tracer Error--
Elevation--1.5 Seconds, 50 Percent Blanking

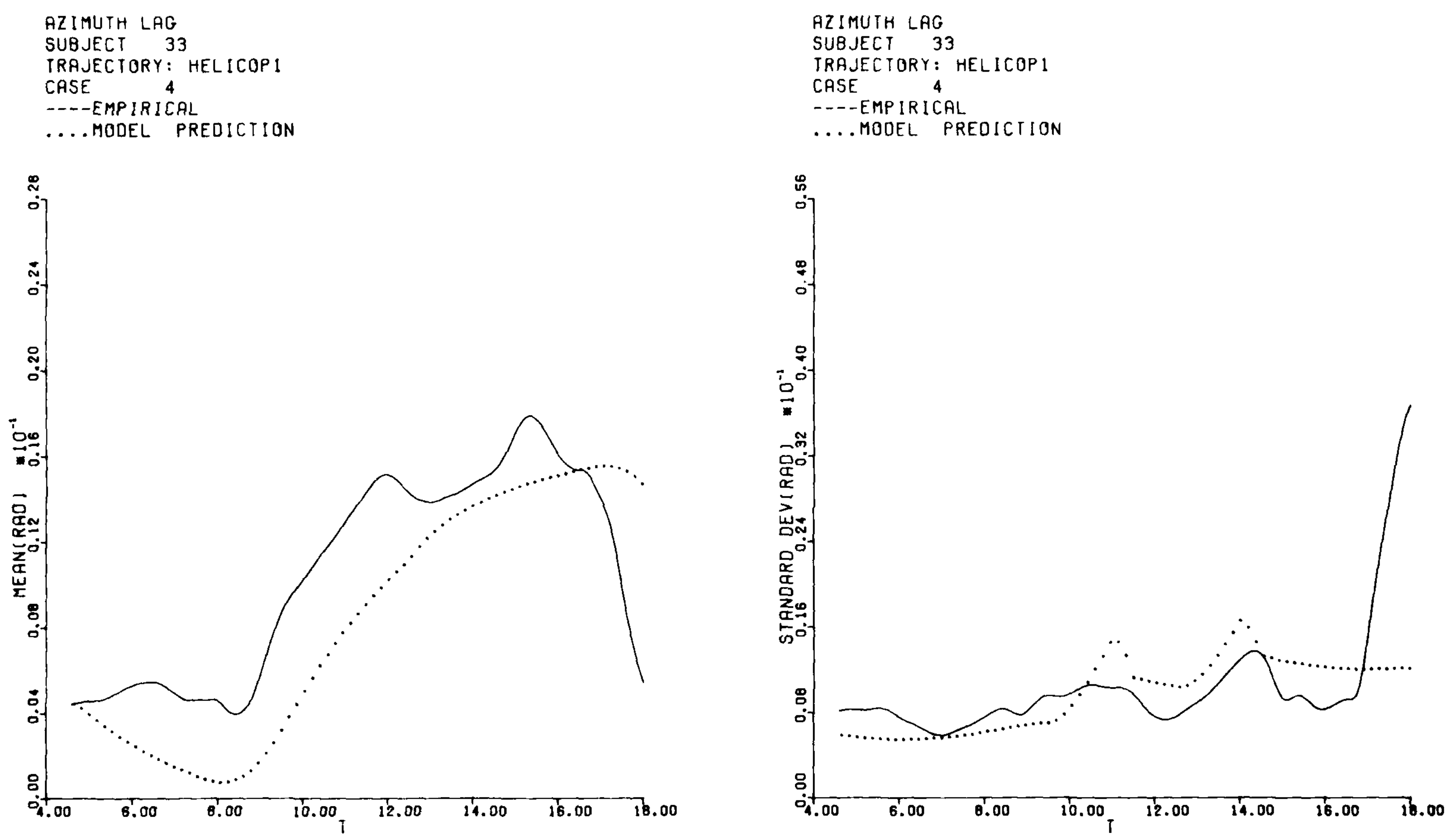


Figure 9a. Mean and Standard Deviation of Tracking Error--
Azimuth--1.5 Seconds, 50 Percent Blanking

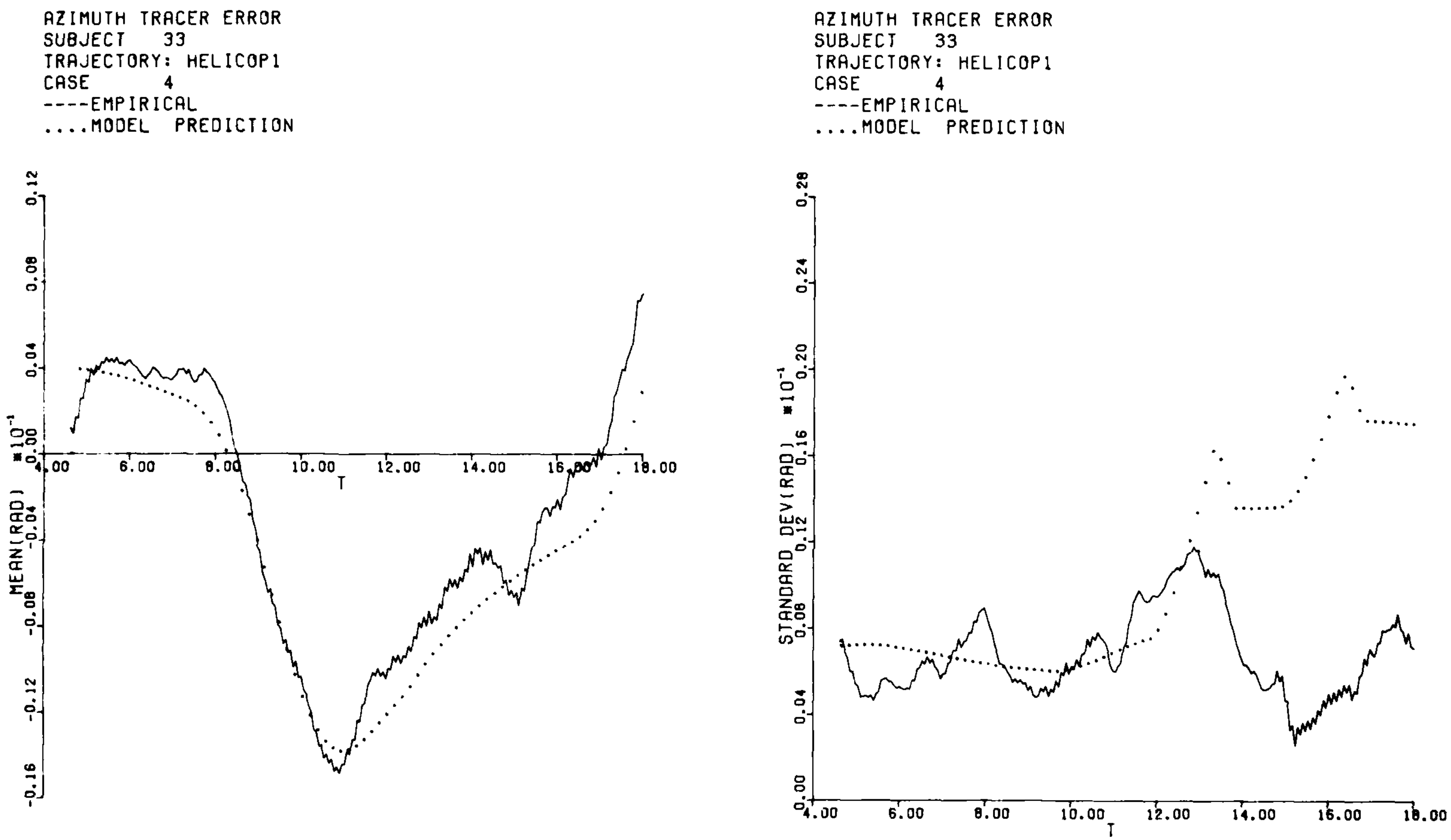


Figure 9b. Mean and Standard Deviation of Tracer Error--
Azimuth--1.5 Seconds, 50 Percent Blanking

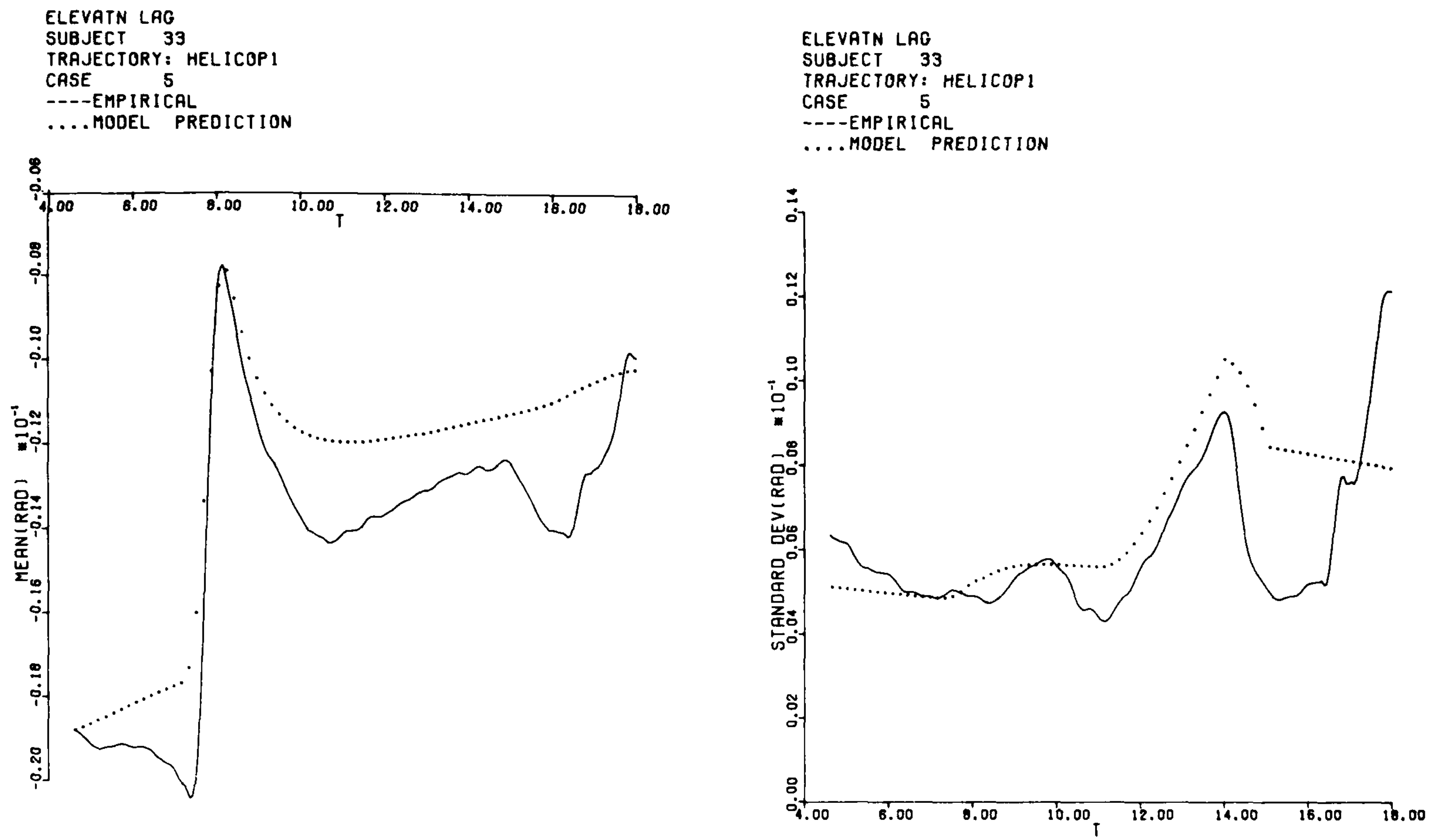


Figure 10a. Mean and Standard Deviation of Tracking Error--Elevation--3.0 Seconds, 50 Percent Blanking

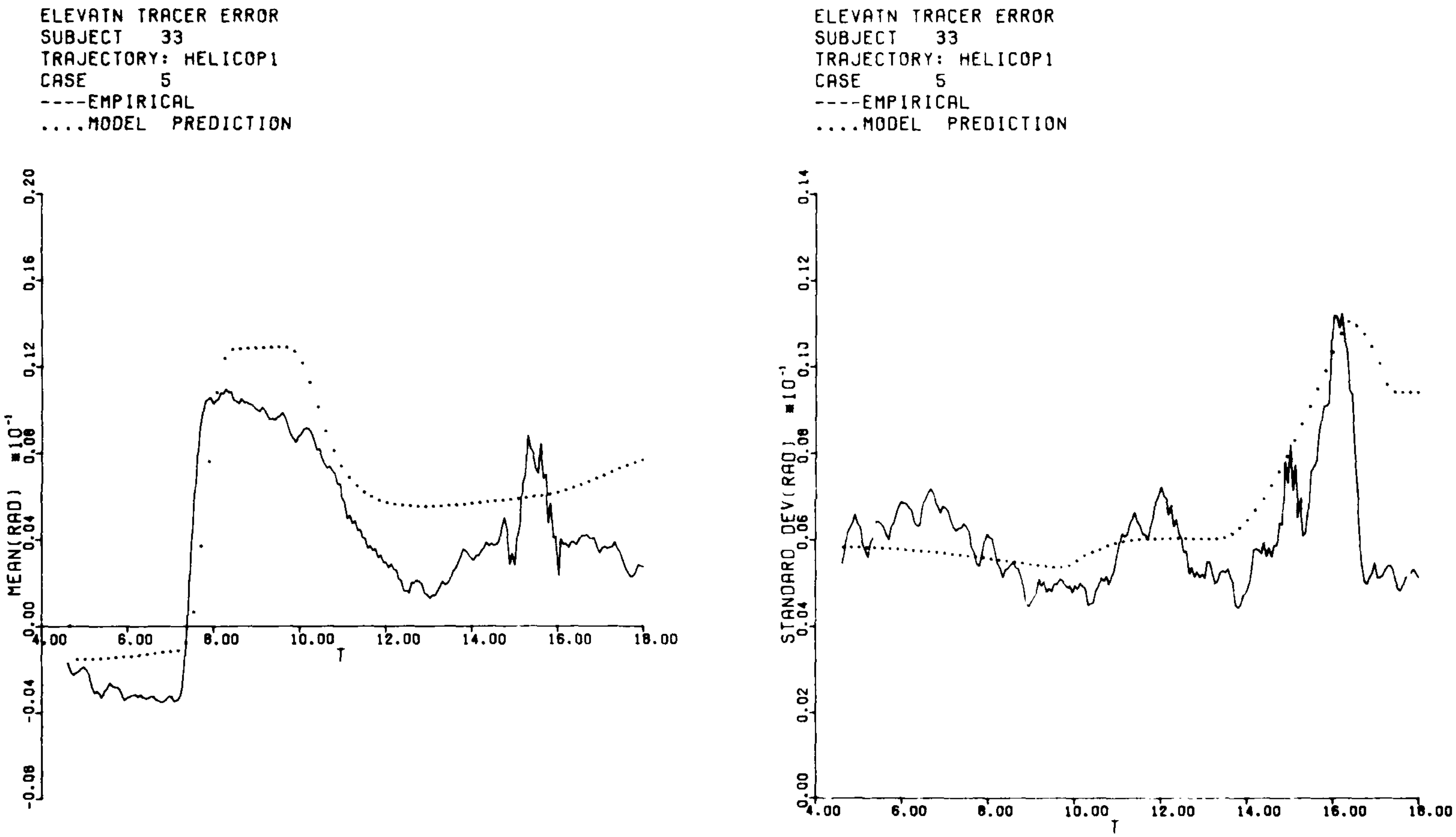


Figure 10b. Mean and Standard Deviation of Tracer Error--
Elevation--3.0 Seconds, 50 Percent Blanking

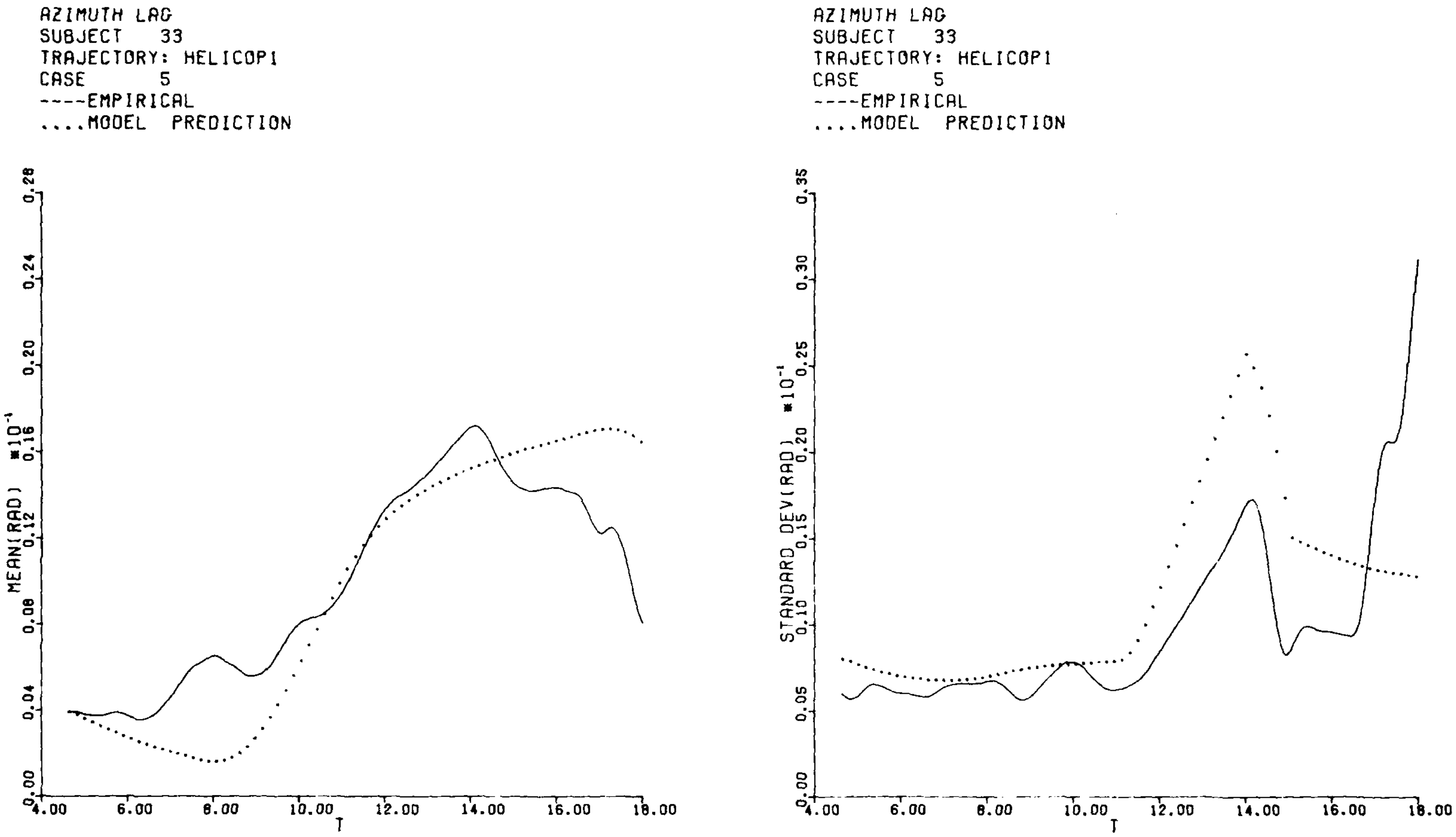


Figure 11a. Mean and Standard Deviation of Tracking Error--
Azimuth--3.0 Seconds, 50 Percent Blanking

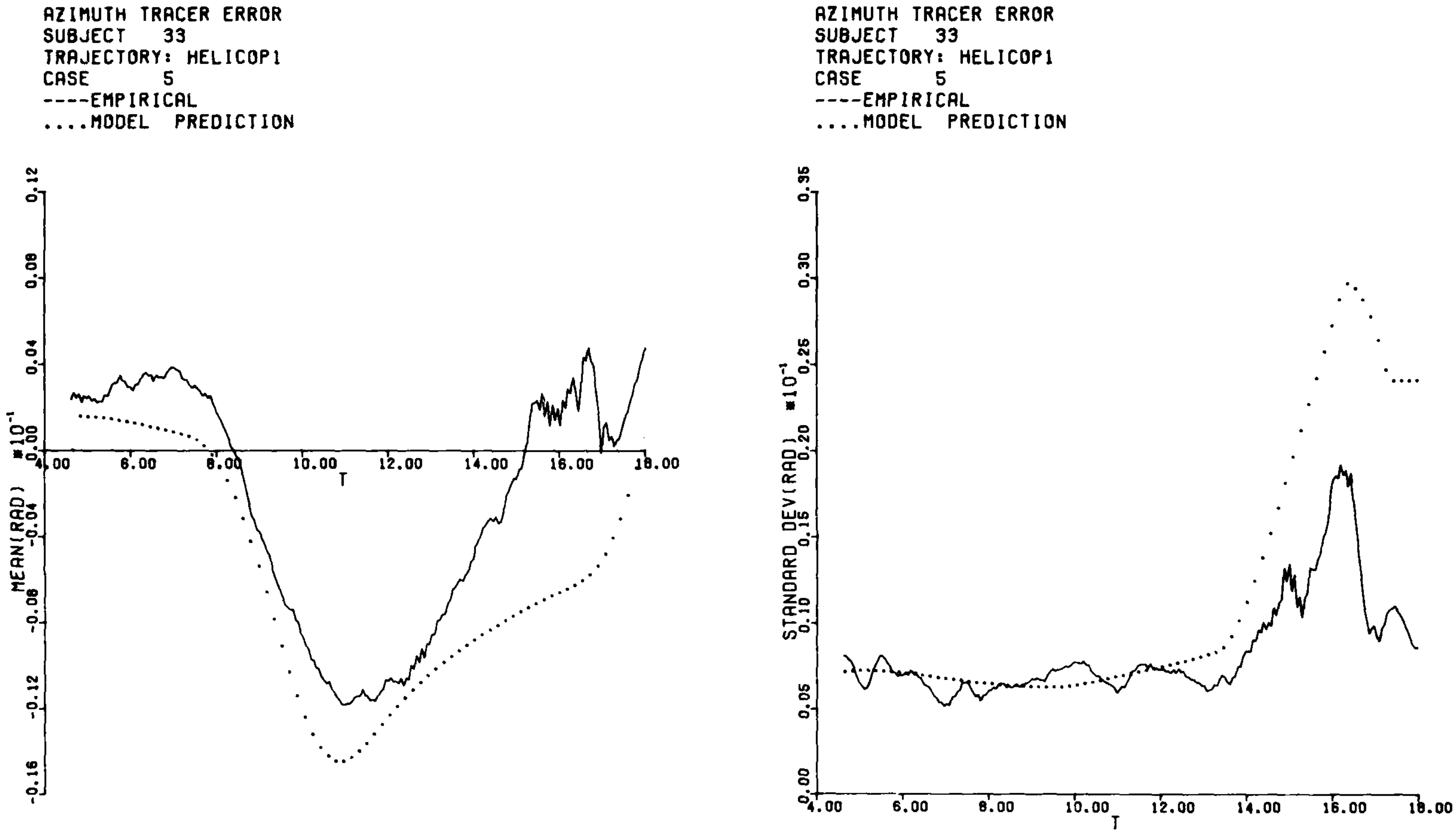


Figure 11b. Mean and Standard Deviation of Tracer Error--
Azimuth--3.0 Seconds, 50 Percent Blanking

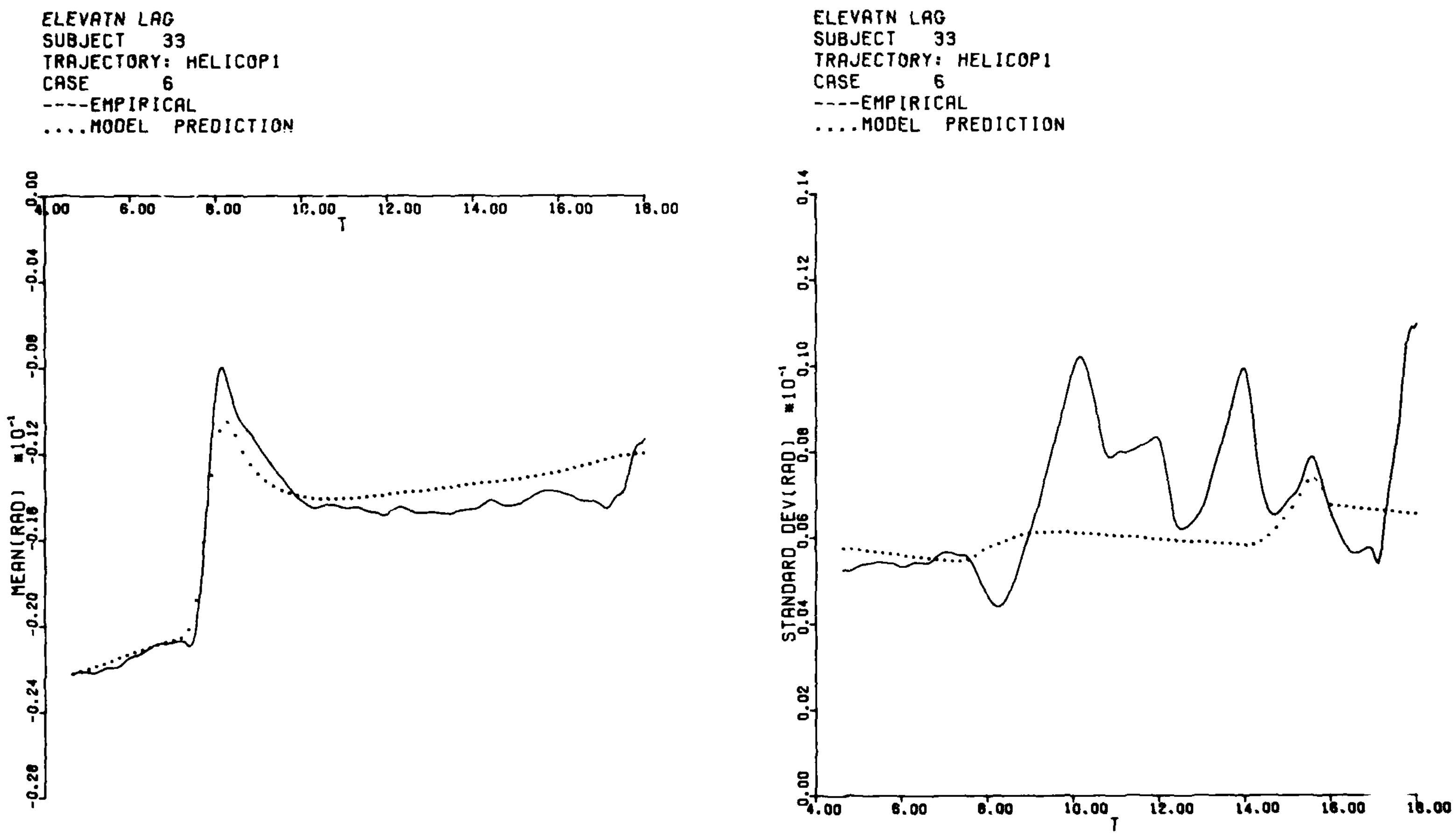
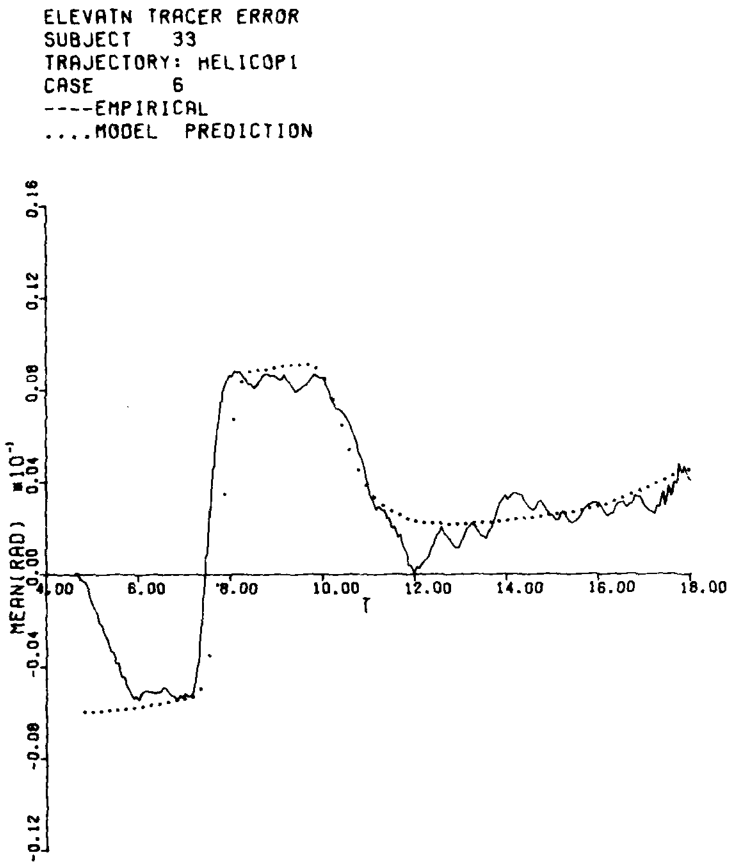


Figure 12a. Mean and Standard Deviation of Tracking Error—Elevation—6.0 Seconds, 50 Percent Blanking



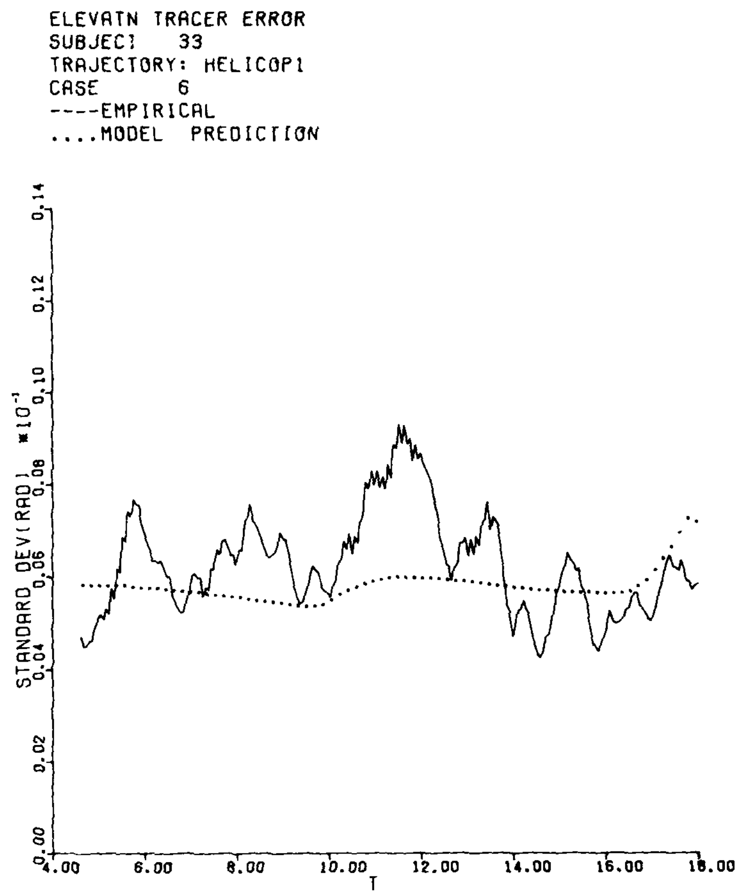


Figure 12b. Mean and Standard Deviation of Tracer Error--
Elevation--6.0 Seconds, 50 Percent Blanking

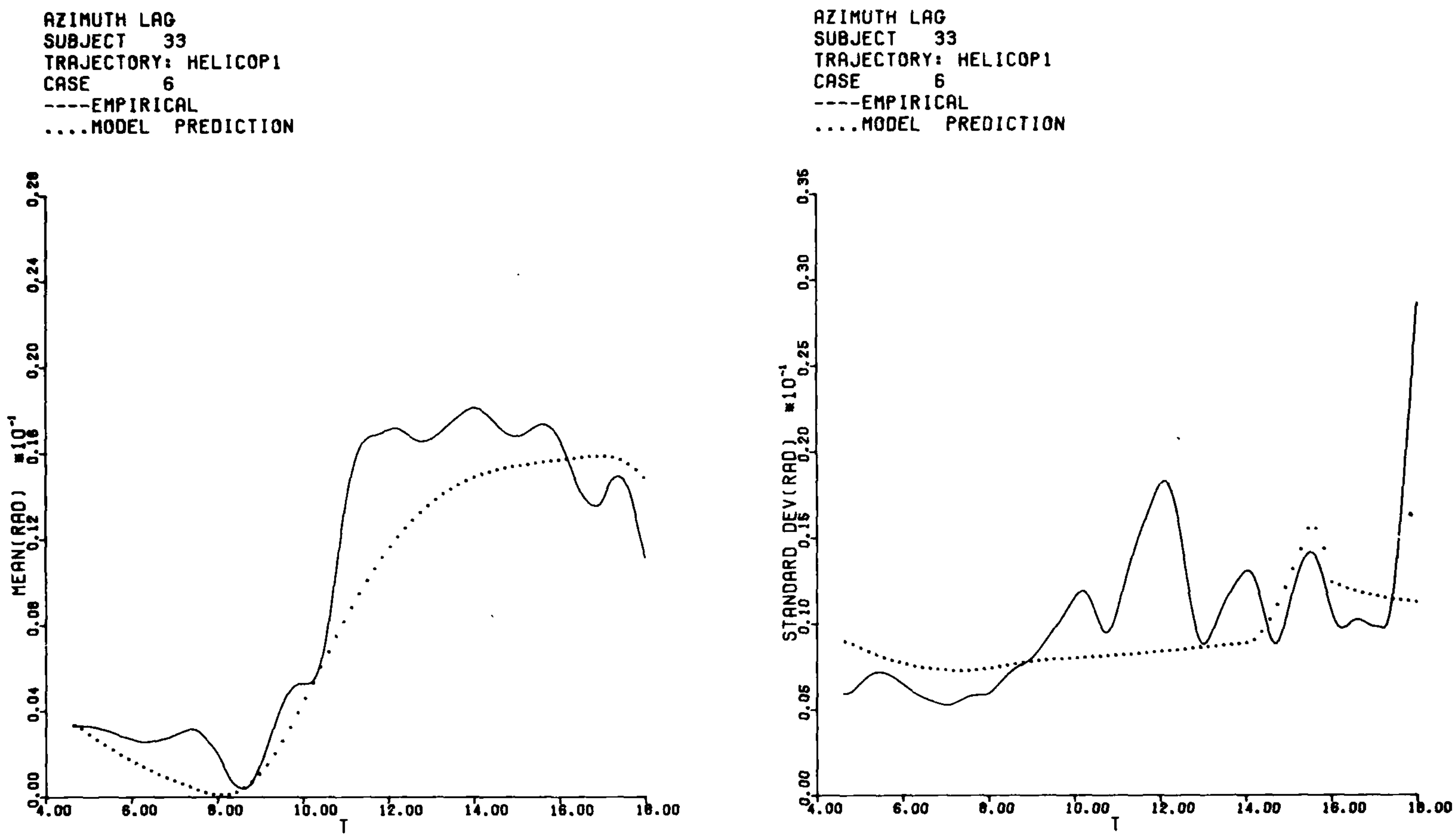
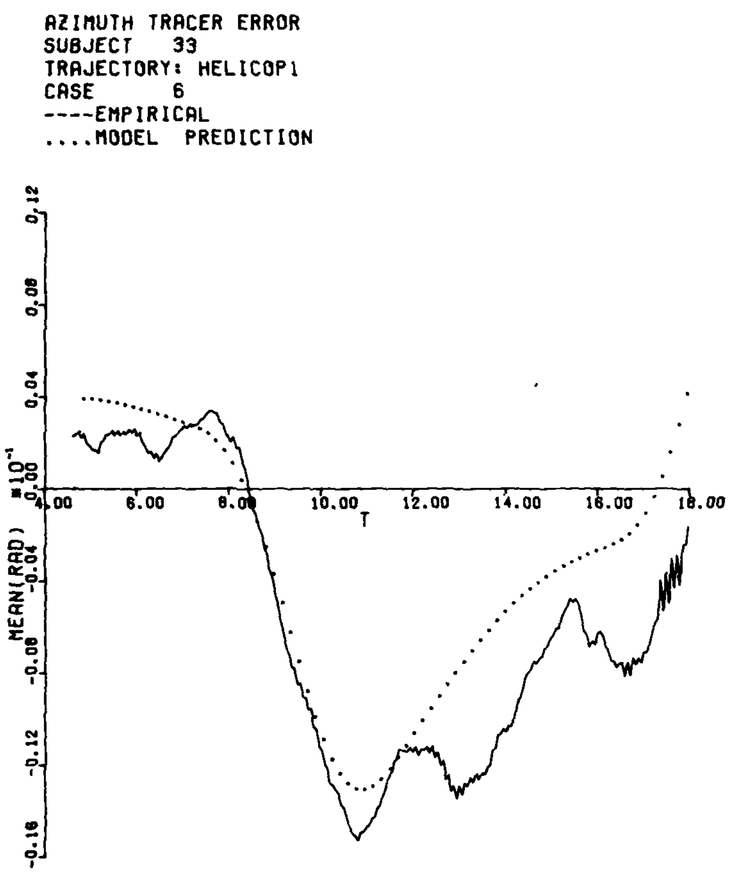


Figure 13a. Mean and Standard Deviation of Tracking Error--Azimuth--6.0 Seconds, 50 Percent Blanking



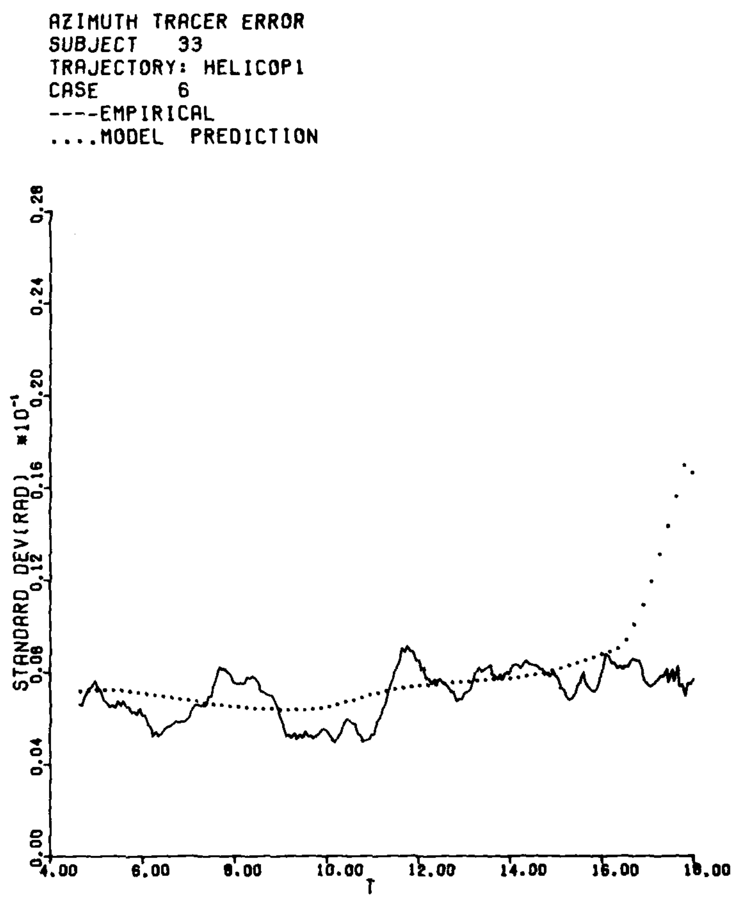


Figure 13b. Mean and Standard Deviation of Tracer Error—Azimuth—6.0 Seconds, 50 Percent Blanking

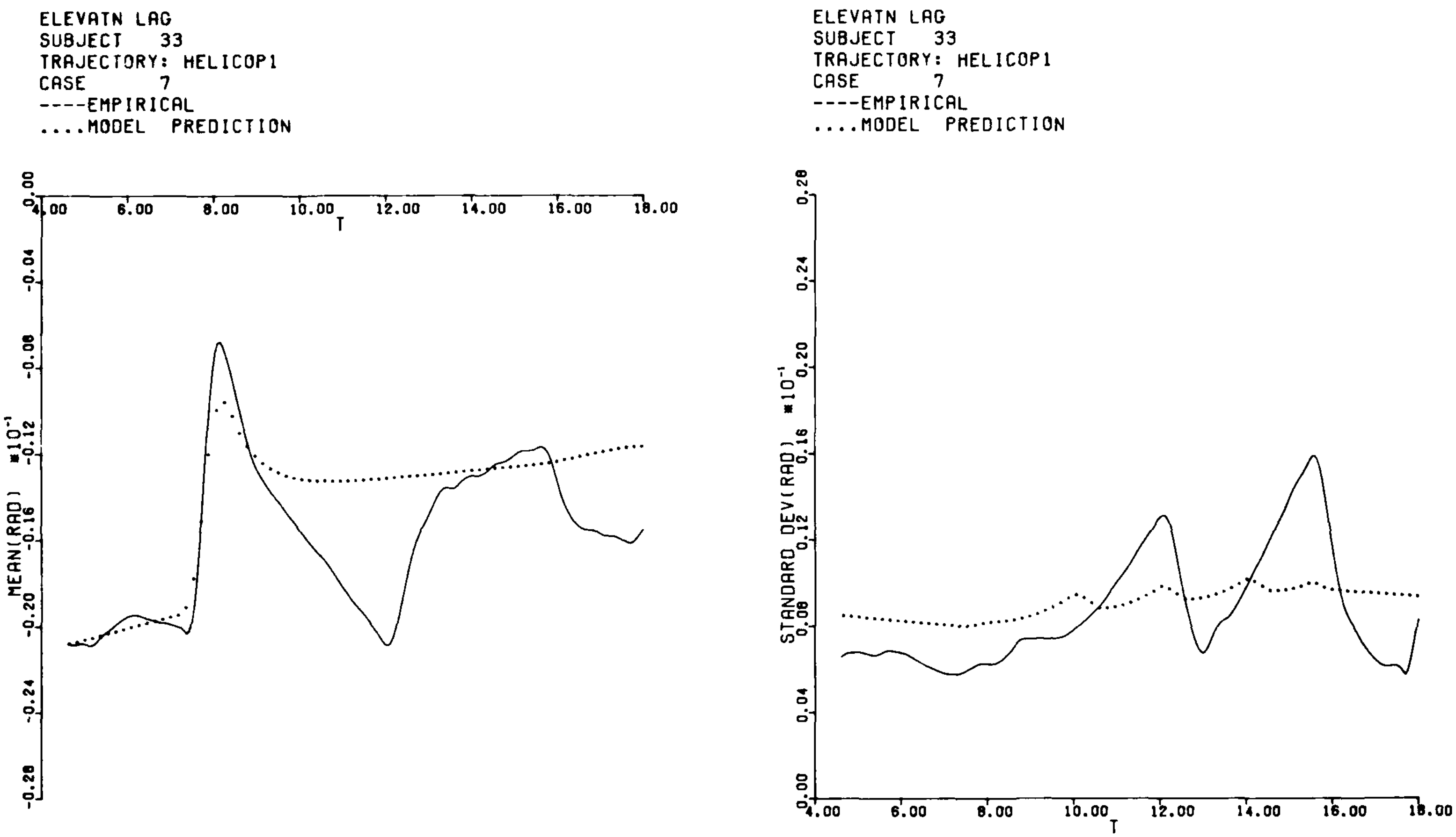
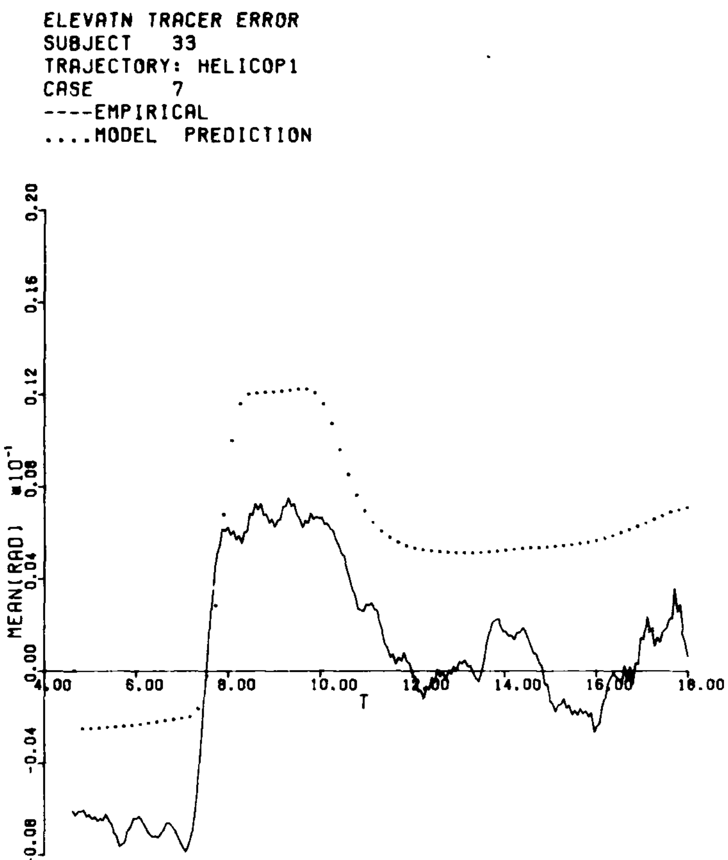


Figure 14a. Mean and Standard Deviation of Tracking Error—Elevation--1.5 Seconds, 75 Percent Blanking



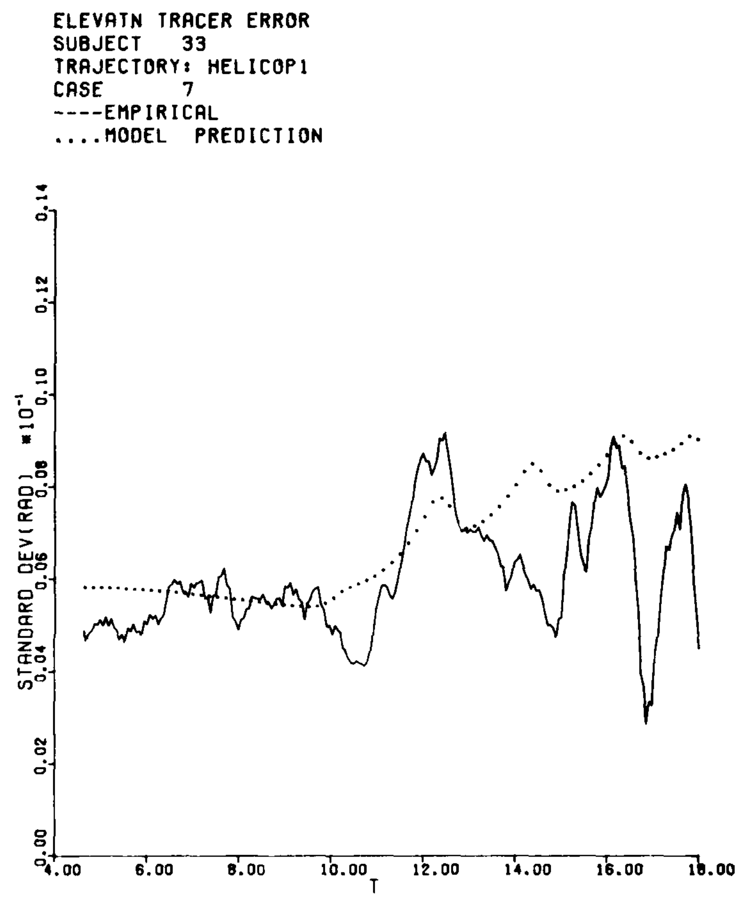


Figure 14b. Mean and Standard Deviation of Tracer Error--
Elevation--1.5 Seconds, 75 Percent Blanking

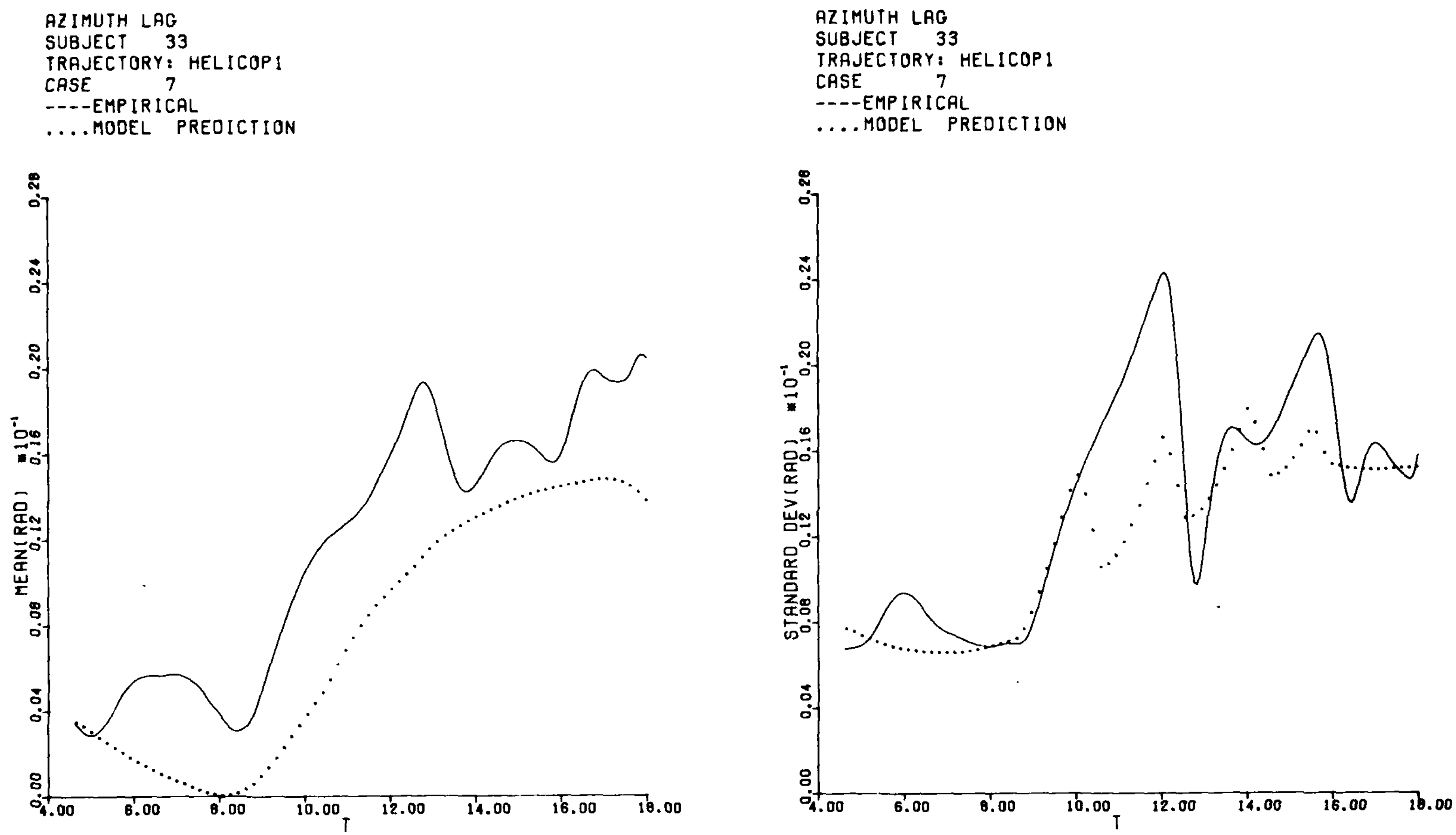
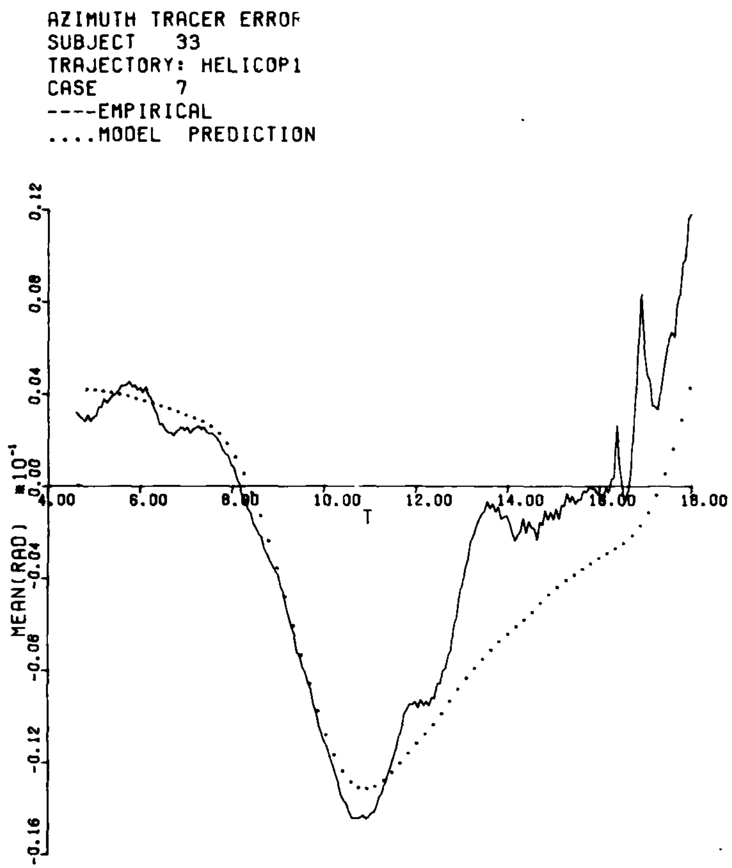


Figure 15a. Mean and Standard Deviation of Tracking Error—Azimuth—1.5 Seconds, 75 Percent Blanking



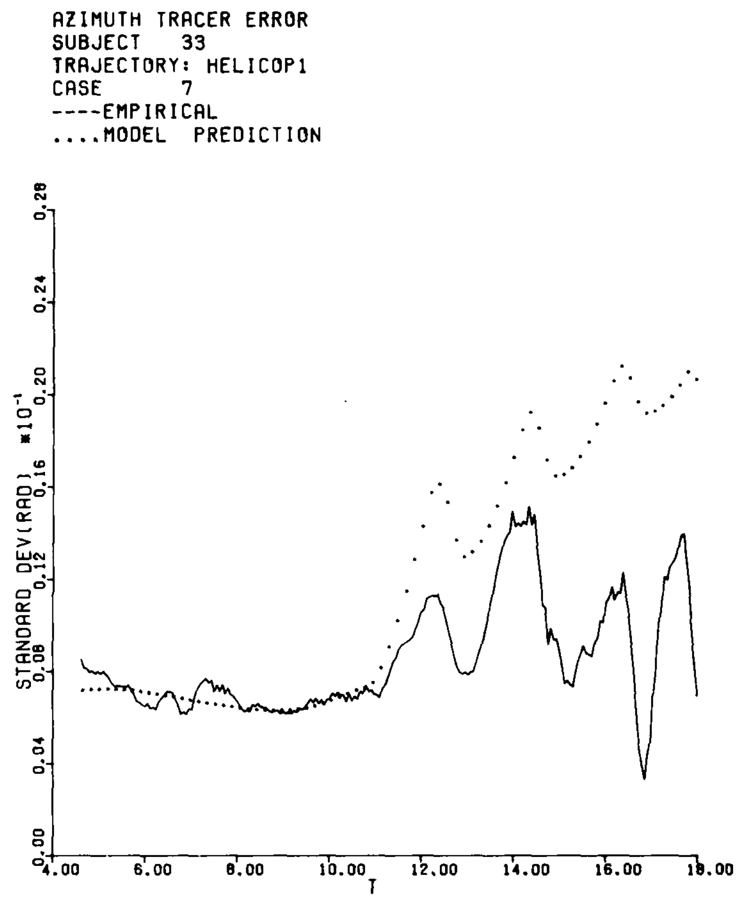
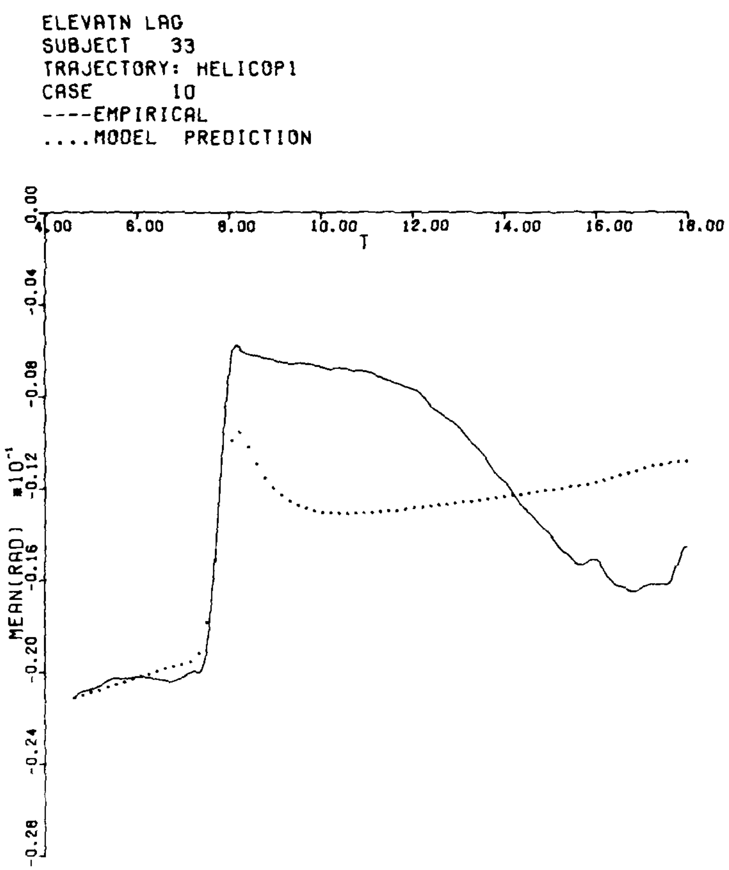


Figure 15b. Mean and Standard Deviation of Tracer Error--Azimuth--1.5 Seconds, 75 Percent Blanking



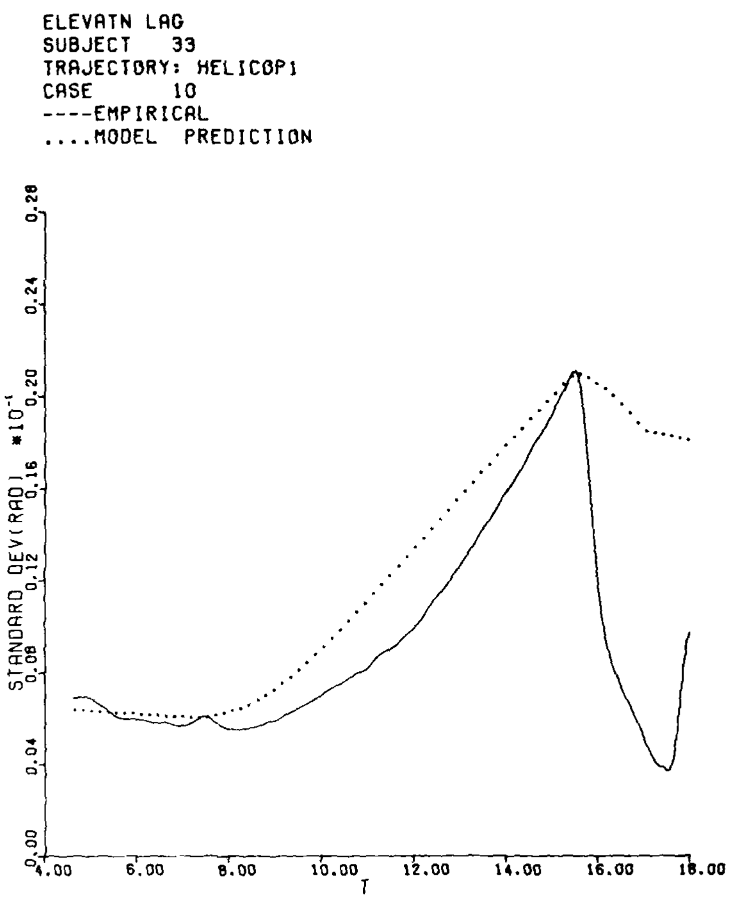


Figure 16a. Mean and Standard Deviation of Tracking Error--
Elevation--1.5 Seconds, 100 Percent Blanking

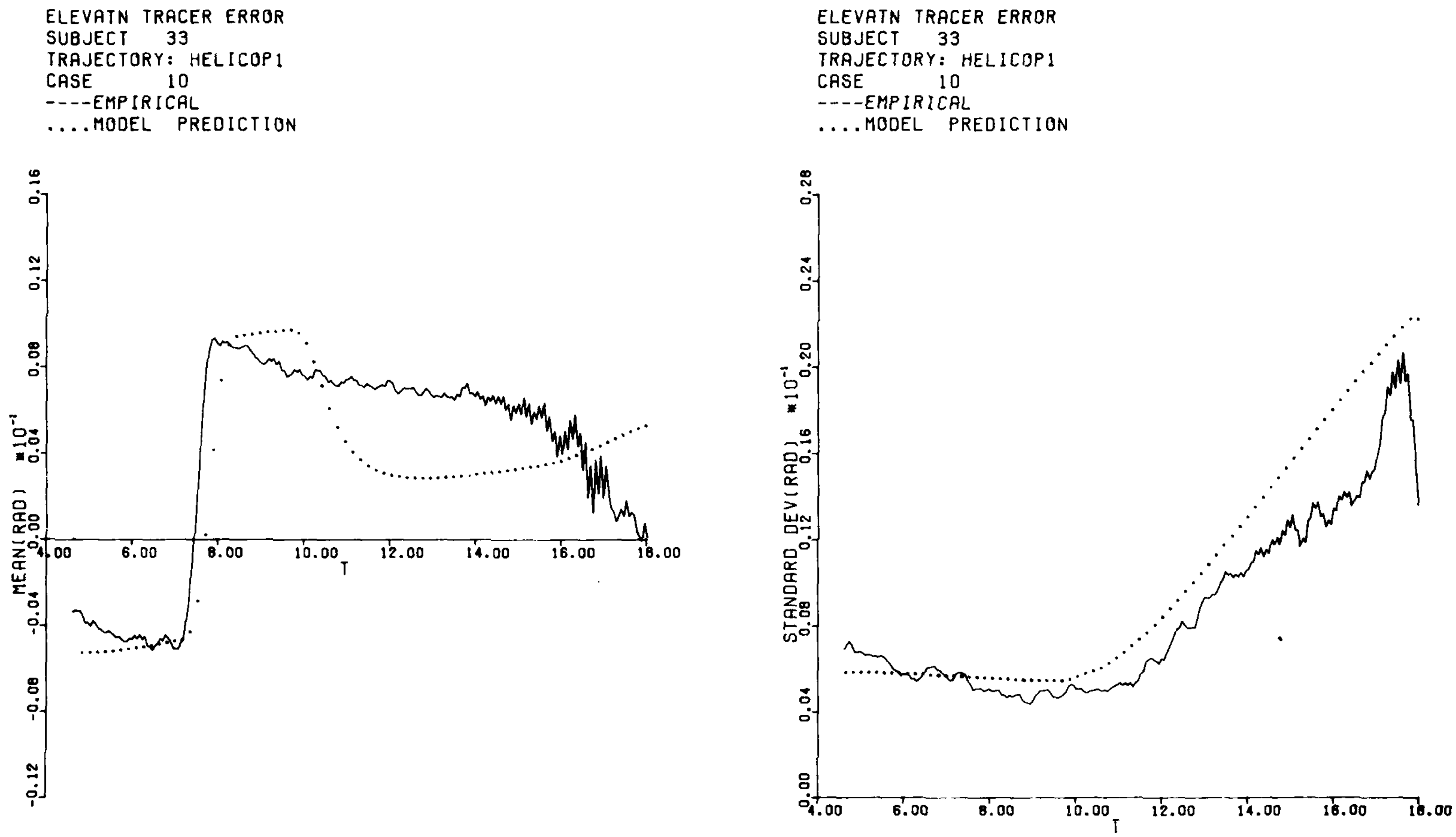


Figure 16b. Mean and Standard Deviation of Tracer Error--Elevation--1.5 Seconds, 100 Percent Blanking

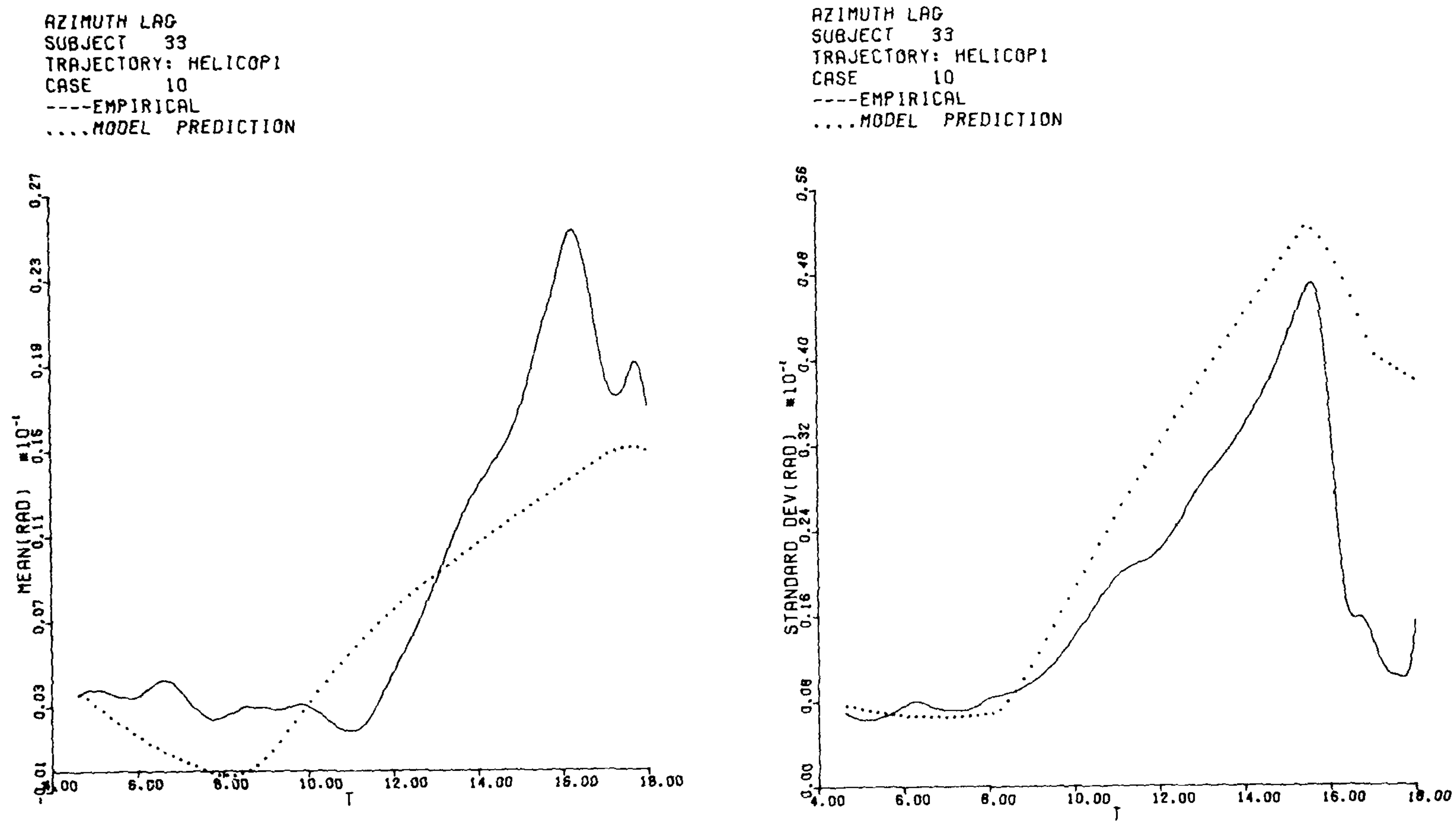


Figure 17a. Mean and Standard Deviation of Tracking Error--
Azimuth--1.5 Seconds, 100 Percent Blanking

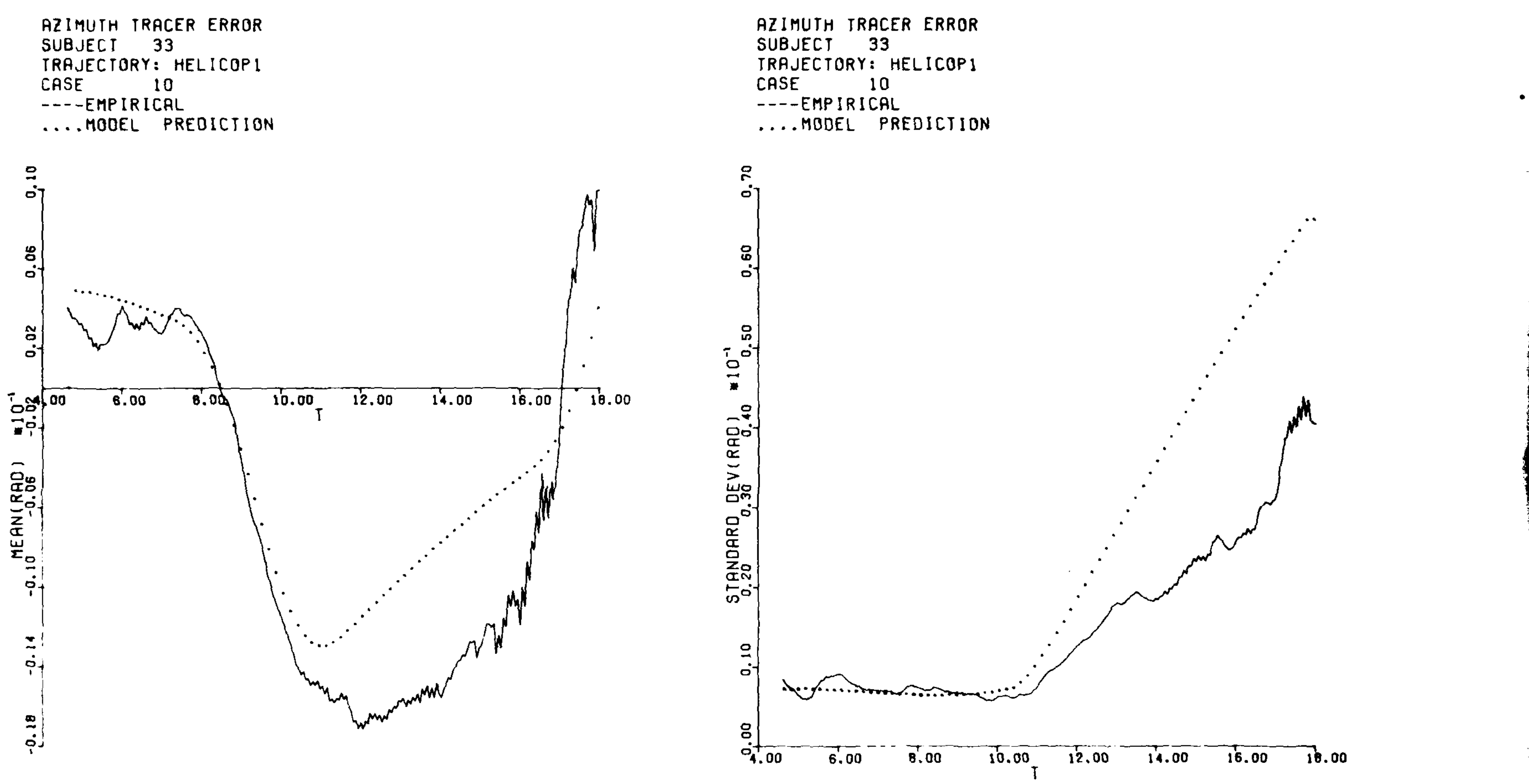


Figure 17b. Mean and Standard Deviation of Tracer Error--
Azimuth--1.5 Seconds, 100 Percent Blanking

Section 5
CONCLUSION

This report summarizes the modeling of a gunner's performance in a complex AAA tracking and firing task under pseudorandom observation interruptions. The highlight of the task is that the gunner fires tracer rounds without the aid of radar and lead angle computer. Furthermore, the gunner's performance is greatly hindered by observation interruptions via blanking the target on an optical display. A blanking model is designed which consists of a reduced-order observer, a linear feedback controller, and a remnant element.

The gunner's performance is parameterized by the controller and estimator gains, in addition to the covariance coefficient of the remnant. The effect of blanking is modeled by degrading these gains and coefficients as a function of blanking duration. An exponential decay form is assumed for these parameters. The associated time constants are determined from empirical data collected in the blanking experiment. A direct search method is used to identify model parameters systematically while minimizing the least-squares error between the model output and the empirical data.

Computer simulation of the proposed gunner model show that the model predictions are in good agreement with empirical data for various blanking patterns using a typical helicopter trajectory. These results demonstrate that the model can adequately describe the gunner's tracking and firing characteristics in an AAA weapon system subject to observation blanking.

This gunner (blanking) model has been incorporated into the MTQ series of P001/OBS AAA engagement models and is designated as program P001/OBS 3/6B. This composite program P001/OBS 3/6B can be used in the evaluation of aircraft survivability and performing weapons effectiveness studies. Documentation of P001/OBS 3/6B is in preparation at the Air Force Aerospace Medical Research Laboratory and will be distributed separately.

APPENDIX A

LISTING OF PARAMETER IDENTIFICATION PROGRAM ELEVATION CASE

```
WOW,T300,CM70300.   L760295,WEI,258-3960
COMMENT.  **OWL6ELID,ID=L760295,CY=1* IDENT EL PARA USE HELI TRAJ*
FTN.
ATTACH,TAPE1,OWL6HELSUBJ33,CY=1,MR=1.
ATTACH,TAPE2,OWL6TRACERSUBJ33,CY=1,MR=1.
LGO.
      PROGRAM OPT(TAPE1,TAPE2,INPUT,OUTPUT)
      DIMENSION ALPHA(7),PSI(7),A(7),D(7),ALPHAN(7)
      DIMENSION SUM5(6),E(7),V(7,7),Z(7,7),XI(7,7),B(7,7),F(6),F2(6,7)
      COMMON/ARRAY/X(1000),S(1000),ELDD(1000),TX(1000),TS(1000),RAN(1000
     1),THET(1000)
      COMMON/S/C0,DEL,NSTP,NDIM,T0,IPT,Y10,Y20
      LOGICAL FH,PAR
      INTEGER Q,SUBIT,FR
      LG=0
      DO 1 I=1,7
      DO 1 J=1,7
      V(I,J)=XI(I,J)=0.0
      IF (I.EQ.J) V(I,J)=XI(I,J)=1.0
      Z(I,J)=0.0
      B(I,J)=0.0
1     CONTINUE
      FR=1
      READ*,K1,NDIM,T0,IPT,Y100,Y10
      PRINT 42,K1,NDIM,T0,IPT,Y10
42    FORMAT(1H1,"NO.OF PTS = ",I4,2X,"ORDER= ",I2,
     C 2X,"INIT TIME= ",G12.5//1X,"READ EVERY ",I2," POINT","Y10= ",G12
     C.5)
      KT=T0/0.03
      DEL=0.03*IPT
      ISET=0
      K=K1-KT
      READ(1,43)(T,DUM6,AZ,AZD,AZDD,EL,ELD,DUM1,AZMN,DUM2,AZSD,DUM3,
     C I=1,KT)
      DO 44 I=1,K
      READ(1,43) T,DUM6,AZ,AZD,AZDD,EL,ELD,DUM1,AZMN,DUM2,AZSD,DUM3
      IF(EOF(1)) 49,45
43    FORMAT(12G12.5)
45    IF(MOD(I-1,IPT).NE.0) GO TO 44
      IH=(I-1)/IPT+1
      IF(IH.EQ.1) EL0=EL
      ELDD(IH)=DUM1
      X(IH)=DUM2
      S(IH)=DUM3
      RAN(IH)=7.5
      THET(IH)=EL
      IF(DUM6.LT.2877.) RAN(IH)=DUM6/(930.-.19*DUM6)
      M=RAN(IH)/DEL
      IF(IH.LE.M) GO TO 44
      ISET=ISET+1
      IF(ISET.NE.1) GO TO 44
      ELTAU=EL
44    CONTINUE
49    CONTINUE
      READ(2,46)(TIME,TAZ,DUM4,TAZSD,DUM5,I=1,KT)
      DO 47 I=1,K
      READ(2,46)TIME,TAZ,DUM4,TAZSD,DUM5
      IF(EOF(2))50,48
46    FORMAT(5G12.5)
48    IF(MOD(I-1,IPT).NE.0) GO TO 47
      IG=(I-1)/IPT+1
      TX(IG)=DUM4
      TS(IG)=DUM5
47    CONTINUE
```

```
50      CONTINUE
        C0=1.34
        NSTP=IH-1
        READ*,(ALPHA(I),I=1,7)
        READ*,EPS
        AJ=0.0
10      DO 8 I=1,7
        E(I)=.1
        D(I)=0.0
        A(I)=2.0
8       CONTINUE
        IT=SUBIT=0
        Y20=ELTAU-(ELO-Y100)+0.001*(5.2*RAN(1)+0.486*RAN(1)**2)*CO
      1 S(ELO-Y100)
        CALL INTG(ALPHA,AJ)
        OLDJ=AJ
        PRINT 40,ALPHA
40      FORMAT("0ALPHA= ",G12.5,":",G12.5,":",G 12.5,":",G12.5,":",
      C G12.5,":",G12.5,":",G12.5)
        PRINT 41,AJ,IT,SUBIT
41      FORMAT(" J= ",G12.5," ITERATIONS= ",I5,"SUBITERATIONS= ",I5)
11      DO 100 K=1,7
        SUBIT=SUBIT+1
        DO 12 L=1,7
        ALPHAN(L)=ALPHA(L)+E(K)*V(K,L)
        IF((L.GE.2).AND.(L.LE.4)) GO TO 12
        IF (ALPHAN(L).LT.0.0) ALPHAN(L)=-ALPHAN(L)
12      CONTINUE
        CALL INTG(ALPHAN,AJ)
        IF (AJ.GT.OLDJ) GO TO 20
        OLDJ=AJ
        PRINT 40,ALPHAN
        PRINT 41,AJ,IT,SUBIT
        D(K)=D(K)+E(K)
        E(K)=3*E(K)
        DO 15 M=1,7
        ALPHA(M)=ALPHAN(M)
15      CONTINUE
        IF (A(K).GT.1.5) A(K)=1.0
        GO TO 25
20      E(K)=-.5*E(K)
        IF (A(K).LE.1.5) A(K)=0.0
25      CK=0.0
        DO 30 L=1,7
        IF (A(L).LT.0.5)GO TO 30
        CK=1.0
30      CONTINUE
        IF (CK.NE.0.0)GO TO 100
        SUM1=SUM2=0.0
        DO 32 M=1,7
        SUM1=SUM1+XI(1,M)**2
        SUM2=SUM2+XI(2,M)**2
32      CONTINUE
        X1=SQRT(SUM1)
        X2=SQRT(SUM2)
        X3=X1/X2
        PRINT 33,OLDJ,ALPHA,X1,X3
33      FORMAT (*1J=     *,G12.5/* ALPHA=(*,G12.5,*:*,G12.5,*:*,G12.5,
      2*:*,G12.5,*:*,G12.5,*:*,G12.5,*:*,G12.5/
      3* XI(1)= *,G12.5/* XI(1)/XI(2)= *,G12.5)
        GO TO 110
100     CONTINUE
105     GO TO 11
110     SUM3=0.0
        DO 115 L=1,6
        DO 115 M=1,7
```

```
      F(L)=0.0
      F2(L,M)=0.0
 115  CONTINUE
      DO 117 J=1,7
      DO 117 K=1,7
 117  XI(J,K)=0.0
      DO 120 I=1,7
 120  SUM3=SUM3+(ABS(D(I)))
      IF (SUM3.LE.EPS)GO TO 1000
      DO 130 N=1,7
      DO 130 J=1,7
      Z(N,J)=D(N)*V(N,J)
 130  CONTINUE
      DO 140 J=1,7
      DO 140 L=1,7
      DO 140 K=J,7
      XI(J,L)=XI(J,L)+Z(K,L)
 140  CONTINUE
      SUM4=0.0
      DO 150 J=1,7
 150  SUM4=SUM4+XI(1,J)**2
      SUM4=SQRT(SUM4)
      DO 155 J=1,7
 155  V(1,J)=XI(1,J)/SUM4
      KOUNT=2
 159  Q=KOUNT-1
      DO 175 K=1,Q
      DO 160 L=1,7
      F(Q)=F(Q) + XI(KOUNT,L)*V(K,L)
 160  CONTINUE
      DO 170 M=1,7
      F2(Q,M)=F(Q)*V(K,M) + F2(Q,M)
 170  CONTINUE
      F(Q)=0.0
 175  CONTINUE
      DO 190 I=1,7
 190  B(KOUNT,I)=XI(KOUNT,I)-F2(Q,I)
      SUM5(Q)=0.0
      DO 200 M=1,7
 200  SUM5(Q)=SUM5(Q)+B(KOUNT,M)**2
      SUM5(Q)=SQRT(SUM5(Q))
      DO 215 M=1,7
      V(KOUNT,M)=B(KOUNT,M)/SUM5(Q)
 215  CONTINUE
      KOUNT=KOUNT+1
      IF (KOUNT.LE.7) GO TO 159
      IT=IT+1
      SUBIT=0
      FR=1
      DO 250 K=1,7
      E(K)=.1
      D(K)=0.0
      A(K)=2.0
 250  CONTINUE
      GO TO 11
 1000 CALL EXIT
      END
      SUBROUTINE INTG(ALPHA,EJ)
      COMMON/ARRAY/XEMP(1000),SEMP(1000),EDD(1000),TX(1000),TS(1000)
     1,TAU(1000),THET(1000)
      COMMON/S/CO,DEL,NSTP,NO,TO,IPT,Y10,Y20
      DIMENSION H(4),P(4,4),P1(4,4),P2(4,4),ALPHA(7),A(4,4)
     1,B(16),EB(16),EBINT(16),EA(4,4),EAINT(4,4),F(4),CQC(4,4),D(4)
     2,X3(1000),X4(1000),EDH(1000),EDDH(1000)
      EQUIVALENCE (A(1,1),B(1)),(EA(1,1),EB(1)),(EAINT(1,1),EBINT(1))
      DATA WT/1./
```

```
C
C INITIALIZATION
C
      SCAL=C0**2/DEL
      DO 1 I=1,ND
      DO 1 J=1,ND
      P(I,J)=0.
      CC(I,J)=0.
1     A(I,J)=0.
      ND1=ND-1
      ND2=ND-2
      NAA=ND/2-1
      DO 11 I=1,ND
      W(I)=0.
      D(I)=0.
      F(I)=0.
11    P(I,I)=0.0000
      P(1,1)=0.0000256279
      P(2,2)=0.0000338677
      X3(1)=0.000
      X4(1)=0.
      EDH(1)=X3(1)
      EDDH(1)=0.
      S1=0.
      ARG=-DEL*ALPHA(1)
      IF(ARG.GT.-200.) S1=EXP(ARG)
C
C COMPUTE AND STORE STATES X3 AND X4
C
C     PRINT 97
97    FORMAT(4X,"TIME",6X,"TARGET VEL",2X,"EST VEL ERROR",2X,"EST TAR VE
     1 L"/)
      DO 10 KK=1,NSTP
C     IF((MOD(KK,100).EQ.0).OR.(KK.EQ.1)) PRINT 96,T,X3(KK),X4(KK)
C    1 ,X3(KK)-X4(KK)
96    FORMAT(4G12.5)
      K1=KK+1
      K2=KK-1
      X3(K1)=X3(KK)+EDD(KK)*DEL
      IF(ALPHA(1).EQ.0.) GO TO 4
      X4(K1)=S1*X4(KK)+EDD(KK)*(1.-S1)/ALPHA(1)
      GO TO 3
4     X4(K1)=X4(KK)+EDD(KK)*DEL
3     CONTINUE
C
C COMPUTE AND STORE ESTIMATED TARGET VELOCITY AND ACCELERATION
C
      EDH(KK)=X3(KK)-X4(KK)
      IF(K2.GE.1) EDDH(KK)=(EDH(KK)-EDH(K2))/DEL
10    CONTINUE
      NDIM=ND
      N1=ND**2
C
C START INTEGRATION LOOP
C
      KP=1
      W(1)=Y10
      W(2)=Y20
      ISET=0
      DO 100 KK=1,NSTP
      M=TAU(KK)/DEL
      IF(KK.LE.M) GO TO 100
      ISET=ISET+1
      IF(ISET.NE.1) GO TO 101
      SPEAN=(W(1)-XEMP(KK))**2+(W(2)-TX(1))**2
      SSD=(SQRT(P(1,1))-SEMP(1))**2+(SQRT(P(2,2))-TS(1))**2
```

```
101   AL1=1.+0.001*(5.2*TAU(KK)+.486*TAU(KK)**2)*SIN(THET(KK-M))
      ALS=ALPHA(2)
      KP=KP+1
      K1=KK+1
      RA=NAA/TAU(KK)
      A2=1.-(TAU(K1)-TAU(KK))/DEL
      COR=C0*ALS
      A(1,1)=-COR
      A(1,2)=-C0*ALPHA(3)
      A(2,ND1)=A(1,1)*AL1*A2
      A(2,ND)=A(1,2)*AL1*A2
      DO 2 I=1,ND2
      J1=I+2
      A(J1,I)=RA
      A(J1,J1)=-RA
2     CONTINUE
      CALL DSORT(NDIM,B,DEL,EB,EBINT,5)
      CR3=C0*ALPHA(4)*AL1*A2
      CR4=C0*ALPHA(4)
      SCAL1=SCAL*(AL1*A2)**2
      CQC(2,2)=ALPHA(5)*SCAL1
C     IF((MOD(KK,100).EQ.0).OR.(KK.EQ.1)) PRINT 99,T,W(1)+Y10,SMEAN,
C    1 SQRT(P(1,1)),SSD
99    FORMAT(5G12.5)
C
C COMPUTE MEAN TRACKING ERROR
C
      DO 110 I=1,NDIM
      DO 120 J=1,NDIM
      D(I)=D(I)+EA(I,J)*W(J)
120   CONTINUE
110   CONTINUE
      F(1)=(1.-CR4)*X3(KK)+CR4*X4(KK)
      F(2)=X3(KK)+CR3*(X4(KK-M)-X3(KK-M))+0.001*(1.-A2)*(5.2+0.972*
     C TAU(KK))*COS(THET(KK-M))
      DO 130 I=1,NDIM
      DO 140 J=1,2
      D(I)=D(I)+EAINT(I,J)*F(J)
140   CONTINUE
      W(I)=D(I)
      D(I)=0.
130   CONTINUE
C
C COMPUTE ERROR DUE TO MEAN TRACKING ERROR
C
      SMEAN=SMEAN+(W(1)-XEMP(KK))**2+(W(2)-TX(KP))**2
C
C COMPUTE COVARIANCE MATRIX
C
      CQC(1,1)=(ALPHA(5)+ALPHA(6)*ABS(EDH(KK))+ALPHA(7)*ABS(EDDH(KK)))
     1 *SCAL
      IF(KK.GT.M) CQC(2,2)=(ALPHA(5)+ALPHA(6)*ABS(EDH(KK-M))
     1 +ALPHA(7)*ABS(EDDH(KK-M)))*SCAL1
      CALL MULT(EAINT,CQC,NDIM,N1,P1,10)
      CALL MULT(EA,P,NDIM,N1,P2,10)
      DO 220 I=1,NDIM
      DO 220 J=1,NDIM
      P(I,J)=P1(I,J)+P2(I,J)
220   CONTINUE
C
C COMPUTE ERROR DUE TO STANDARD DEVIATION
C
      SSD=SSD+(SQRT(P(1,1))-SEMP(KK))**2+(SQRT(P(2,2))-TS(KP))**2
100   CONTINUE
      EJ=DEL*(SMEAN+WT*SSD)
      RETURN
```

```
      END
      SUBROUTINE MULT(E,F,L,L1,H,MR)
      DIMENSION E(1),F(1),G(15),H(1)
      DO 10 I=1,L
      II=I
      DO 10 K=1,L
      TEMP=0.
      DO 5 J=I,L1,L
      TEMP=TEMP+E(J)*F(II)
 5    II=II+1
      KK=(K-1)*L+I
      H(KK)=TEMP
 10   G(KK)=TEMP
      IF(MR.EQ.1)RETURN
      DO 20 I=1,L
      DO 20 K=I,L
      TEMP=0.
      II=K
      DO 15 J=I,L1,L
      TEMP=TEMP+G(J)*E(II)
 15   II=II+L
      KK=(K-1)*L+I
 20   H(KK)=TEMP
      L2=L-1
      DO 30 I=1,L2
      L3=I+1
      DO 30 J=L3,L
      K1=(I-1)*L+J
      K2=(J-1)*L+I
 30   H(K1)=H(K2)
      END
      SUBROUTINE DSCRT(NDIM,A,DEL,EA,EAINT,NT)
      DIMENSION A(1),EA(1),EAINT(1),COEF(30)
C        SETS EA=EXP(A*DEL),EAINT=INTEGRAL EA 0 TO DEL
      NDIM1=NDIM+1
      NN=NDIM*NDIM
      NTM1=NT-1
      COEF(NT)=1.
      DO 10 I=1,NTM1
      II=NT-I
   10 COEF(II)=DEL*COEF(II+1)/FLOAT(I)
C             NT MUST BE AT LEAST 3
      CALL DIAG(NDIM,EAINT,A,COEF(1),COEF(2))
      DO 60 L=3,NT
      CALL MULT(A,EAINT,NDIM,NN,EA,1)
      IF(L.EQ.NT)GO TO 70
   60 CALL DIAG(NDIM,EAINT,EA,1.0,COEF(L))
   70 DO 80 II=1,NN,NDIM1
      EA(II)=EA(II)+1.0
   80 CONTINUE
      END
      SUBROUTINE DIAG(NDIM,A,B,C1,C2)
      DIMENSION A(1),B(1)
      NDIM1=NDIM+1
      NN=NDIM*NDIM
      NM1=NDIM-1
      II=1
      IF(C1.EQ.1.0) GO TO 10
      DO 5 J=1,NN,NDIM
      K=J+NM1
      DO 4 I=J,K
    4 A(I)=C1*B(I)
      A(II)=A(II)+C2
    5 II=II+NDIM1
      RETURN
   10 DO 7 J=1,NN,NDIM
```

```
      K=J+NM1
      DO 6 I=J,K
    6 A(I)=B(I)
      A(II)=A(II)+C2
    7 II=II+NDIM1
      RETURN
      END
535,4,2.46,2,-0.017964,-0.020511
1.5431,.017491,.024433,.42318,.22446E-7,.17975E-3,.17302E-3
0.01
```

APPENDIX B

LISTING OF PARAMETER IDENTIFICATION PROGRAM AZIMUTH CASE

```
WOW,T300,CM75100.   L760295,WEI,258-3960
MAP(ON).
COMMENT.  "OWL6AZIC,ID=L760295,CY=1; IDENT AZ PARA USE WEL1 TRAJ"
FIN.
ATTACH,TAPE1,OWL6HELSUBJ33,CY=1,MR=1.
ATTACH,TAPE2,OWL6TRACERSUBJ33,CY=1,ID=L760295,MR=1.
LGO.
      PROGRAM OPT(TAPE1,TAPE2,INPUT,OUTPUT)
      DIMENSION ALPHA(7),PSI(7),A(7),D(7),ALPHAN(7)
      DIMENSION SUMS(6),E(7),V(7,7),Z(7,7),XI(7,7),B(7,7),F(6),F2(6,7)
      COMMON/ARRAY/ X(1000),S(1000),AZDD(1000),ELS(1000),RAN(1000)
     1,TX(1000),TS(1000)
      COMMON/S/CO,DEL,NSTP,NDIM,T0,IPT,Y10,Y20,AZ30
      LOGICAL FM,PAR
      INTEGER D,SUBIT,FR
      LG=0
      NPAR=7
      NPAR1=NPAR-1
      DO 1 I=1,NPAR
      DO 1 J=1,NPAR
      V(I,J)=XI(I,J)=0.0
      IF (I.EQ.J)  V(I,J)=XI(I,J)=1.0
      Z(I,J)=0.0
      B(I,J)=0.0
1     CONTINUE
      FR=1
      READ*,K1,NDIM,T0,IPT,Y100,Y10
      PRINT 42,K1,NDIM,T0,IPT,Y10
42    FORMAT(1H1,"NO.OF PTS =",I4,2X,"ORDER= ",I2,
     C 2X,"INIT TIME= ",G12.5//1X,"READ EVERY ",I2," POINT",",Y10= ",G12
     C.5)
      KT=T0/0.03
      K=K1-KT
      DEL=0.03*IPT
      ISET=0
      READ(1,43)(T,DUM4,AZ,AZD,DUM1,EL,ELD,ELDD,DUM2,ELMN,DUM3,ELSD,
     C I=1,KT)
      DO 44 I=1,K
      READ(1,43) T,DUM4,AZ,AZD,DUM1,EL,ELD,ELDD,DUM2,ELMN,DUM3,ELSD
      IF(EOF(1)) 44,45
43    FORMAT(12G12.5)
45    IF(MOD(I-1,IPT).NE.0) GO TO 44
      IH=(I-1)/IPT+1
      IF(IH.NE.1) GO TO 49
      AZ0=AZ
      AZD0=AZD
49    CONTINUE
      AZDD(IH)=DUM1
      X(IH)=DUM2
      ELS(IH)=EL-ELMN
      S(IH)=DUM3
      RAN(IH)=7.5
      IF(DUM4.LT.2877.) RAN(IH)=DUM4/(930.-.19*DUM4)
      M=RAN(IH)/DEL
      IF(IH.LE.M) GO TO 44
      ISET=ISET+1
      IF(ISET.NE.1) GO TO 44
      AZTAU=AZ
44    CONTINUE
      READ(2,46)(TIME,DUM5,TEL,DUM7,TELSD,ID=1,KT)
      DO 47 I=1,K
      READ(2,46) TIME,DUM5,TEL,DUM7,TELSD
      IF(EOF(2)) 48,48
46    FORMAT(5G12.5)
```

```
46    IF(MOD(I-1,IPT).NE.0.) GO TO 47
      IG=(I-1)/IPT+1
      TX(IG)=DUM5
      TS(IG)=DUM7
47    CONTINUE
      C0=1.28
      NSTP=IH-1
      READ*,(ALPHA(I),I=1,NPAR)
      READ*,EPS
      PRINT*,EPS,(ALPHA(I),I=1,NPAR)
      AJ=0.0
10    DO 8 I=1,NPAR
      E(I)=.1
      D(I)=0.0
      A(I)=2.0
8     CONTINUE
      IT=SUBIT=0
      Y20=AZTAU-(AZ0-Y100)
      CALL INTG(ALPHA,AJ)
      OLDJ=AJ
      PRINT 40,(ALPHA(MP),MP=1,NPAR)
40    FORMAT("0ALPHA= ",G12.5,":",G12.5,":",G12.5,":",G12.5,":",
     C G12.5,":",G12.5,":",G12.5)
      PRINT 41,AJ,IT,SUBIT
41    FORMAT(" J= ",G12.5," ITERATIONS= ",I5,"SUBITERATIONS= ",I5)
11    DO 100 K=1,NPAR
      SUBIT=SUBIT+1
      DO 12 L=1,NPAR
      ALPHAN(L)=ALPHA(L)+E(K)*V(K,L)
      IF((L.GE.2).AND.(L.LE.4)) GO TO 12
      IF (ALPHAN(L).LT.0.0) ALPHAN(L)=-ALPHAN(L)
12    CONTINUE
      CALL INTG(ALPHAN,AJ)
      IF (AJ.GT.OLDJ) GO TO 20
      OLDJ=AJ
      PRINT 40,(ALPHAN(MP),MP=1,NPAR)
      PRINT 41,AJ,IT,SUBIT
      D(K)=D(K)+E(K)
      E(K)=3*E(K)
      DO 15 M=1,NPAR
      ALPHA(M)=ALPHAN(M)
15    CONTINUE
      IF (A(K).GT.1.5) A(K)=1.0
      GO TO 25
20    E(K)=-.5*E(K)
      IF (A(K).LE.1.5) A(K)=0.0
25    CK=0.0
      DO 30 L=1,NPAR
      IF (A(L).LT.0.5)GO TO 30
      CK=1.0
30    CONTINUE
      IF (CK.NE.0.0)GO TO 100
      SUM1=SUM2=0.0
      DO 32 M=1,NPAR
      SUM1=SUM1+XI(1,M)**2
      SUM2=SUM2+XI(2,M)**2
32    CONTINUE
      X1=SQRT(SUM1)
      X2=SQRT(SUM2)
      X3=X1/X2
      PRINT 33,OLDJ,(ALPHA(MP),MP=1,NPAR),X1,X3
33    FORMAT (*1J=     *,G12.5/* ALPHA=(*,G12.5,*:*,G12.5,*:*,G12.5,
     2*:*,G12.5,*:*,G12.5,*:*,G12.5,*:*,G12.5,*)*/
     3* XI(1)= *,G12.5/* XI(1)/XI(2)= *,G12.5)
      GO TO 110
100   CONTINUE
```

```
105   GO TO 11
110   SUM3=0.0
      DO 115 L=1,NPAR1
      DO 115 M=1,NPAR
      F(L)=0.0
      F2(L,M)=0.0
115   CONTINUE
      DO 117 J=1,NPAR
      DO 117 K=1,NPAR
117   XI(J,K)=0.0
      DO 120 I=1,NPAR
120   SUM3=SUM3+(ABS(D(I)))
      IF (SUM3.LE.EPS)GO TO 1000
      DO 130 N=1,NPAR
      DO 130 J=1,NPAR
      Z(N,J)=D(N)*V(N,J)
130   CONTINUE
      DO 140 J=1,NPAR
      DO 140 L=1,NPAR
      DO 140 K=J,NPAR
      XI(J,L)=XI(J,L)+Z(K,L)
140   CONTINUE
      SUM4=0.0
      DO 150 J=1,NPAR
150   SUM4=SUM4+XI(1,J)**2
      SUM4=SQRT(SUM4)
      DO 155 J=1,NPAR
155   V(1,J)=XI(1,J)/SUM4
      KOUNT=2
159   Q=KOUNT-1
      DO 175 K=1,Q
      DO 160 L=1,NPAR
      F(Q)=F(Q) + XI(KOUNT,L)*V(K,L)
160   CONTINUE
      DO 170 M=1,NPAR
      F2(Q,M)=F(Q)*V(K,M) + F2(Q,M)
170   CONTINUE
      F(Q)=0.0
175   CONTINUE
      DO 190 I=1,NPAR
190   B(KOUNT,I)=XI(KOUNT,I)-F2(Q,I)
      SUM5(Q)=0.0
      DO 200 M=1,NPAR
200   SUM5(Q)=SUM5(Q)+B(KOUNT,M)**2
      SUM5(Q)=SQRT(SUM5(Q))
      DO 215 M=1,NPAR
      V(KOUNT,M)=B(KOUNT,M)/SUM5(Q)
215   CONTINUE
      KOUNT=KOUNT+1
      IF (KOUNT.LE.NPAR) GO TO 159
      IT=IT+1
      SUBIT=0
      FR=1
      DO 250 K=1,NPAR
      E(K)=.1
      D(K)=0.0
      A(K)=2.0
250   CONTINUE
      GO TO 11
1000  CALL EXIT
      END
      SUBROUTINE INTG(ALPHA,EJ)
      COMMON/ARRAY/XEMP(1000),SEMP(1000),EDD(1000),ELG(1000),TAU(1000)
     1,TX(1000),TS(1000)
      COMMON/S/CO,DEL,NSTP,ND,TO,IPT,Y10,Y20,AZ00
      DIMENSION W(4),P(4,4),P1(4,4),P2(4,4),ALPHA(7),A(4,4)
```

```
      1,E(16),EB(16),EBINT(16),EA(4,4),EAINT(4,4),F(4),CQC(4,4),D(4)
      2,X3(1000),X4(1000),EDH(1000),EDDH(1000)
       EQUIVALENCE  (A(1,1),B(1)),(EA(1,1),EB(1)),(EAINT(1,1),EBINT(1))
       DATA WT/1./
C
C INITIALIZATION
C
       DO 1 I=1,ND
       DO 1 J=1,ND
       P(I,J)=0.
       CQC(I,J)=0.
1      A(I,J)=0.
       NDIV=NSTP/3
       ND1=ND-1
       ND2=ND-2
       NAA=ND/2-1
       DO 11 I=1,ND
       W(I)=0.
       D(I)=0.
       F(I)=0.
11     P(I,I)=0.0000
       ISET=0
       X3(1)=AZDD
       X4(1)=0.
       EDH(1)=X3(1)
       EDDH(1)=0.
       NDIM=ND
       N1=ND**2
       S1=0.
       DO 10 KK=1,NSTP
       K1=KK+1
       CB=COS(ELG(KK))
       ARG=-DEL*ALPHA(1)*CB
       IF(ARG.GT.-200.) S1=EXP(ARG)
       X3(K1)=X3(KK)+EDD(KK)*DEL
       IF(ALPHA(1).EQ.0.) GO TO 4
       X4(K1)=S1*(X4(KK)+EDD(KK)*DEL)
       GO TO 3
4      X4(K1)=X4(KK)+EDD(KK)*DEL
3      CONTINUE
       EDH(K1)=X3(K1)-X4(K1)
       IF(KK.GE.2)EDDH(KK)=(EDH(KK)-EDH(KK-1))/DEL
       M=TAU(KK)/DEL
       IF(KK.LE.M) GO TO 10
       ISET=ISET+1
       IF(ISET.NE.1) GO TO 10
       W(1)=Y1*CB
       W(2)=Y2*CB
       P(1,1)=(0.0059083*CB)**2
       P(2,2)=(0.0071587*CB)**2
       SMEAN=(W(1)/CB-XEMP(1))**2+(W(2)/CB-TX(1))**2
       SSD=(SQRT(P(1,1))/CB-SEMP(1))**2+(SQRT(P(2,2))/CB-TS(1))**2
       IST=KK
10     CONTINUE
       T=T0
       DO 100 KK=IST,NSTP
       K1=KK+1
       M=TAU(KK)/DEL
       RA=NAA/TAU(KK)
       DO 2 I=1,ND2
       J1=I+2
       A(J1,I)=RA
       A(J1,J1)=-RA
2      CONTINUE
       THEBD=(ELG(K1)-ELG(KK))/DEL
       CB=COS(ELG(KK))
```

```
      TB=-THE3D*TAN(ELG(KK))
      SCAL=(C0*CB)**2/DEL
      T=T+DEL
      A2=1.-(TAU(K1)-TAU(KK))/DEL
      COR=C0*ALPHA(2)*CB
      A(1,1)=-COR+TB
      A(1,2)=-C0*ALPHA(3)*CB
      A(2,ND1)=-COR*A2
      A(2,ND)=A(1,2)*A2
      A(2,2)=TB
C
C COMPUTE TRANSITION MATRIX EA AND ITS INTEGRAL EAINT
C
      CALL DSCRT(NDIM,B,DEL,EB,EBINT,5)
      CR4=C0*ALPHA(4)*CB
C
C START INTEGRATION LOOP
C
C     IF((MOD(KK,10).EQ.0).OR.(KK.EQ.1)) PRINT 99,T,W(1),SMEAN,
C    1 SQRT(P(1,1)),SSO
99    FORMAT(5G12.5)
C
C COMPUTE MEAN TRACKING ERROR
C
      DO 110 I=1,NDIM
      DO 120 J=1,NDIM
      D(I)=D(I)+EA(I,J)*W(J)
120   CONTINUE
110   CONTINUE
      F(1)=(C3-CR4)*X3(KK)+CR4*X4(KK)
      F(2)=X3(KK)*CB
      IF(KK.GT.M) F(2)=F(2)+CR4*(X4(KK-M)-X3(KK-M))*A2
      DO 130 I=1,NDIM
      DO 140 J=1,2
      D(I)=D(I)+EAINT(I,J)*F(J)
140   CONTINUE
      W(I)=D(I)
      D(I)=0.
130   CONTINUE
C
C COMPUTE ERROR DUE TO MEAN TRACKING ERROR
C
      SMEAN=SMEAN+(W(1)/CB-XEMP(K1))**2+(W(2)/CB-TX(K1-IST))**2
C
C COMPUTE COVARIANCE MATRIX
C
      COC(1,1)=(ALPHA(5)+ALPHA(6)*ABS(EDH(KK))+ALPHA(7)*ABS(EDDH(KK)))
     1 *SCAL
      COC(2,2)=ALPHA(5)*SCAL*A2**2
      IF(KK.GT.M) COC(2,2)=(ALPHA(5)+ALPHA(6)*ABS(EDH(KK-M))+ALPHA(7)*
     1 ABS(EDDH(KK-M)))*SCAL*A2**2
      CALL MULT(EAINT,COC,NDIM,N1,P1,10)
      CALL MULT(EA,P,NDIM,N1,P2,10)
      DO 220 I=1,NDIM
      DO 220 J=1,NDIM
      P(I,J)=P1(I,J)+P2(I,J)
220   CONTINUE
C
C COMPUTE ERROR DUE TO STANDARD DEVIATION
C
      SSO=SSO+(SQRT(P(1,1))/CB-SEMP(K1))**2+(SQRT(P(2,2))/CB-TS(K1-IST))
     A**2
100   CONTINUE
      EJ=DEL*(SMEAN+WT*SSO)
      RETURN
      END
```

```
      SUBROUTINE MULT(E,F,L,L1,H,MR)
      DIMENSION E(1),F(1),G(16),H(1)
      DO 10 I=1,L
      II=1
      DO 10 K=1,L
      TEMP=0.
      DO 5 J=I,L1,L
      TEMP=TEMP+E(J)*F(II)
    5 II=II+1
      KK=(K-1)*L+I
      H(KK)=TEMP
   10 G(KK)=TEMP
      IF(MR.EQ.1)RETURN
      DO 20 I=1,L
      DO 20 K=I,L
      TEMP=0.
      II=K
      DO 15 J=I,L1,L
      TEMP=TEMP+G(J)*E(II)
   15 II=II+L
      KK=(K-1)*L+I
   20 H(KK)=TEMP
      L2=L-1
      DO 30 I=1,L2
      L3=I+1
      DO 30 J=L3,L
      K1=(I-1)*L+J
      K2=(J-1)*L+I
   30 H(K1)=H(K2)
      END
      SUBROUTINE DSCRT(NDIM,A,DEL,EA,EAINT,NT)
      DIMENSION A(1),EA(1),EAINT(1),COEF(30)
C        SETS EA=EXP(A*DEL),EAINT=INTEGRAL EA 0 TO DEL
      NDIM1=NDIM+1
      NN=NDIM*NDIM
      NTM1=NT-1
      COEF(NT)=1.
      DO 10 I=1,NTM1
      II=NT-I
   10 COEF(II)=DEL*COEF(II+1)/FLOAT(I)
C           NT MUST BE AT LEAST 3
      CALL DIAG(NDIM,EAINT,A,COEF(1),COEF(2))
      DO 60 L=3,NT
      CALL MULT(A,EAINT,NDIM,NN,EA,1)
      IF(L.EQ.NT)GO TO 70
   60 CALL DIAG(NDIM,EAINT,EA,1.0,COEF(L))
   70 DO 80 II=1,NN,NDIM1
      EA(II)=EA(II)+1.0
   80 CONTINUE
      END
      SUBROUTINE DIAG(NDIM,A,B,C1,C2)
      DIMENSION A(1),B(1)
      NDIM1=NDIM+1
      NN=NDIM*NDIM
      NM1=NDIM-1
      II=1
      IF(C1.EQ.1.0) GO TO 10
      DO 5 J=1,NN,NDIM
      K=J+NM1
      DO 4 I=J,K
    4 A(I)=C1*B(I)
      A(II)=A(II)+C2
    5 II=II+NDIM1
      RETURN
   10 DO 7 J=1,NN,NDIM
      K=J+NM1
```

```
      DO 6 I=J,K
    6 A(I)=B(I)
      A(II)=A(II)+C2
    7 II=II+NDIM1
      RETURN
      END
535,4,2.46,2,0.001304,0.0026046
1.,1.,1.,1.,.00001,.00001,.0001
0.01
```

APPENDIX C

LISTING OF AN AAA GUNNER MODEL SIMULATION PROGRAM

```
M8M,T20,CM70000. L760!95,MEI,2393960
COMMENT.*NEWOWLSTMU6,ID=L760295,CY=1*
COMMENT. *AAA MODE6 BLANKING SIMJLATION PROGRAM*
ATTACH,TAPE1,OWL64ELSUBJ33,ID=L760295,CY=1,MR=1.
FTN.
LGO.
      PROGRAM SIMU6(INPJT,OUTPJT,TAPE1)
      COMMON/S/C0(2),DEL,IH,NOIM,Y10(2),X3(2),EL,ELDD,AZDD,MTAU,RA,A2,ND
     A1,ND2,NAA,UEL,UAZ,ELTR,AZTR,ISET,Z1,Z2,TAU,TS(15),TE(15),T,IBL
C
C THE PURPOSE OF THIS PROGRAM IS TO SIMULATE AN ELEVTN AND AZIMUTH TRACK
C TASK IN THE TRACER-DIRECTED FIRE (MODE 6) SYSTEM
C SUBJECT TO OPTICAL BLANKING
C INPUT: THE ELEVATION (EL) & AZIMUTH (AZ) ANGULAR ACCELERATION OF
C TARGET, AND BLANKING DURATIONS(UP TO 15) IN CHRONOLOGICAL
C ORDER
C OUTPUT: MEAN AND STAND DEV OF LAG ANGLE
C ***** ALL ANGLES ARE IN UNITS OF RADIAN *****
C TAU: DELAY IN SECONDS
C ALPHA: PARAMETER VECTOR
C ELERR: MEAN EL LAG ANGLE (I.E. TARGET ANGLE-BARREL ANGLE)
C AZERR: MEAN AZ LAG ANGLE
C ELSD: STANDARD DEVIATION OF ELEVATION LAG ANGLE
C AZSD: STANDARD DEVIATION OF AZ LAG ANGLE
C ELTR: MEAN EL TRACER ERROR (TARGET ANGLE-TRACER ENDING ANGLE)
C AZTR: MEAN AZ TRACER ERROR
C ELBAR: MEAN EL BARREL ANGLE
C DEL: TIME STEP USED IN THE INTEGRATION ROUTINE
C TS(I): STARTING TIME OF I-TH BLANKING DURATION
C TE(I): ENDING TIME OF I-TH BLANKING DURATION
C Y10(1): INITIAL GUESS OF EL LAG ANGLE
C Y102: INITIAL GUESS OF AZ LAG ANGLE
C UEL: EL CONTROL
C UAZ: AZ CONTROL
C C0(1): EL RATE CONTROL COEFF
C C0(2): AZ RATE CONTROL COEFF
C K1: NO OF POINTS IN THE ENTIRE TRAJECTORY
C K: NO OF POINTS AFTER THE FIRST TRACER ROUND IS FIRED
C ELDD: EL ANGULAR ACCELERATION OF TARGET
C AZDD: AZ ANGULAR ACCELERATION OF TARGET
C X3(1): EL ANGULAR VELOCITY OF TARGET
C X3(2): AZ ANGULAR VELOCITY OF TARGET
C X4: ESTIMATION ERROR OF ANGULAR VELOCITY OF TARGET
C EL: EL ANGULAR POSITION OF TARGET
C H(1): MODEL PREDICTED LAG ANGLE
C H(2): MODEL PREDICTED TRACER ERROR
C P(1,1): VARIANCE OF PREDICTED LAG ANGLE
C P(2,2): VARIANCE OF PREDICTED TRACER ERROR
C T0: THE INITIAL FIRING TIME
C
      READ*,K1,T0,IPT,IBL
      PRINT 3,K1,T0,IPT
3     FORMAT(1H1,"NO OF PTS= ",I4,2X,"INIT TIME= ",G12.5//1X,"READ EVERY
     C",I2," POINT"/)
      IF(IBL.GT.0) READ*,(TS(K),TE(K),K=1,IBL)
      PRINT 11,IBL
11    FORMAT(1X,I5,1X,"BLANKING INTERVALS ARE "/)
      IF(IBL.GE.1) PRINT 4,(TS(K),TE(K),K=1,IBL)
4     FORMAT(5(1X,"(",F9.2,",",F9.2,")"))
      KT=T0/0.03
      K=K1-KT
      T=T0
      NOIM=4
```

```
      IPRINT=20/IPT
      C0(1)=1.34
      C0(2)=1.28
      ND1=NDIM-1
      ND2=NDIM-2
      NAA=NDIM/2-1
      IST=1
      DEL=0.03*IPT
      ELSD=.005**0.5
      AZSD=.005**0.5
      Z1=0.
      Z2=0.
      ISET=0
      UEL=0.
      UAZ=0.
      PRINT 7
 7    FORMAT(/1H ,2X,"TIME",9X,"EL VEL",9X,"ELERR ",5X,"ELSD",6X,
     1"EL CTR",6X,"AZ VEL",6X,"AZERR",6X,"AZSD",6X,"AZ CTR"
     2,6X,"EL TR",6X,"AZ TR"/)
      READ(1,2)(T,DUM1,AZ,AZD,AZDD,EL,ELD,ELDD,AZMN,X,ZSD,S,I=1,KT
     C )
      DO 5 I=1,K
      READ(1,2)T,DUM1,AZ,AZD,AZDD,EL,ELD,ELDD,AZMN,X,ZSD,S
      IF(EOF(1))1,1
 1    IF(MOD(I-1,IPT).NE.0) GO TO 5
      IH=(I-1)/IPT+1
      T=T0+(IH-1)*DEL
      TAU=7.5
      IF(DUM1.LE.2877.) TAU=DUM1/(930.-.19*DUM1)
      MTAU=TAU/DEL
      IF(IST.EQ.1) OTAU=TAU
      IST=IST+1
      RA=NAA/TAU
      A2=1.-(TAU-OTAU)/DEL
      OTAU=TAU
      X3(1)=ELD
      X3(2)=AZD
      IF((IH-1).GE.MTAU) GO TO 9
      TAUR=TAU
C     IF(DUM1.LE.4400) TAUR=TAU*AMAX1(0.6,DUM1/5000.)
      Y10(1)=-TAUR*ELD-.001*(5.2*TAUR+.485*TAUR**2)*COS(EL+0.05)
C     Y102=-0.025*SIGN(1.,AZD)
      Y102=-TAUR*AZD
      IF(IH.NE.1) GO TO 10
      EL0=EL
      AZ0=AZ
      E10=Y10(1)
      E20=Y102
      GO TO 10
 9    ISET=ISET+1
      IF(ISET.NE.1) GO TO 10
      Z1=EL-(EL0-E10)+.001*(5.2*TAU+.486*TAU**2)*COS(EL0+0.05)
      Z2=AZ-(AZ0-E20)
 10   CALL OBSEL6(ELERR,ELSD)
      ELBAR=EL-ELERR
      IF((IH-1).LE.MTAU) Y10(2)=Y102*COS(ELBAR)
      CALL OBSAZ5(AZERR,AZSD,ELBAR)
      IF((MOD(IH-1,IPRINT).EQ.0).OR.(IH.EQ.1)) PRINT 6,T,X3(1),ELERR
     1,ELSD,UEL,X3(2),AZERR,AZSD,UAZ,ELTR,AZTR
 6    FORMAT(11G12.5)
 5    CONTINUE
 2    FORMAT(12G12.5)
      STOP
      END
      SUBROUTINE OBSEL6(ELERR,ELSD)
      COMMON/S/C0(2),DEL,KK,ND,Y10(2),X30(2),EL,EDD,AZDD,M,RA,A2
```

```
     A,ND1,ND2,N44,U,UAZ,ELTR,AZTR,ISET,Z1,Z2,TAU,TS(15),TE(15),T,IBL
      DIMENSION W(4),P(4,4),P1(4,4),P2(4,4),ALPHA(7),A(4,4)
     1,B(16),EB(16),EBINT(16),EA(4,4),EAINT(4,4),F(4),CQC(4,4),D(4)
     2,X3(1000),X4(1000),EDH(1000),EDDH(1000),THET(1000),ALP5D(1000)
      EQUIVALENCE (A(1,1),B(1)),(EA(1,1),EB(1)),(EAINT(1,1),EBINT(1))
      DATA WT/1./
      DATA ALPHA/1.5471,.017491,.024433,.42318,.22446E-7,.17975E-3,
     A .17302E-3/
C
C INITIALIZATION
C
      IF((KK-1).GE.M) GO TO 6
      IF(KK.GT.1) GO TO 5
      N1=ND**2
      NDIM=ND
      ALP1=ALP10=ALPHA(1)
      ALP2=ALPHA(2)
      ALP3=ALPHA(3)
      ALP4=ALPHA(4)
      ALP5=ALPHA(5)
      ALP6=ALPHA(5)
      ALP7=ALPHA(7)
      ALP5D(1)=ALP5
      ACC=0.
      IFLAG=0
      SCAL=CO(1)**2/DEL
      DO 1 I=1,ND
      DO 1 J=1,ND
      P(I,J)=0.
      CQC(I,J)=0.
1     A(I,J)=0.
      DO 11 I=1,ND
      W(I)=0.
      D(I)=0.
      F(I)=0.
11    P(I,I)=0.0030
      P(1,1)=0.0000256279
      P(2,2)=0.0000339677
      X3(1)=X30(1)
      X4(1)=0.
      EDH(1)=X3(1)
      EDDH(1)=0.
      S1=0.
5     W(1)=Y10(1)
6     IF(ISET.EQ.1) W(2)=Z1
      IF(IBL.LT.1) GO TO 18
      ALP5=ALPHA(5)
      IS=IFLAG+1
      DO 12 I=IS,IBL
      IF(T.GE.TS(I).AND.T.LT.TE(I)) GO TO 15
      ATT=AMIN1(1.5,ACC/3.)
      IF(T.GE.TE(I).AND.T.LT.(TE(I)+ATT)) GO TO 16
      IF(T.GE.(TE(I)+ATT)) GO TO 21
      ACC=0.
      IF(T.LT.TS(IS)) GO TO 19
      GO TO 12
16    ALP1=ALP10+(ALPHA(1)-ALP10)*(1.-EXP(-0.43*(T-TE(I))))
      ALP5=-0.0001*(1.-EXP(-0.43*(T-TE(I))))
      GO TO 18
15    ACC=ACC+DEL
      IFLAG=I-1
      ALP1=ALPHA(1)*EXP(-0.0755*(T-TS(I)))
      ALP3=ALPHA(3)*EXP(-0.52*(T-TS(I)))
      ALP5=0.0001*(1.-EXP(-0.12*(T-TS(I))))
      ALP10=ALP1
      GO TO 18
```

```
21     ALP1=ALPHA(1)
       ALP3=ALPHA(3)
       ALP5=ALPHA(5)
       GO TO 18
12     CONTINUE
18     CONTINUE
       ALP5D(KK)=ALP5
       ARG=-DEL*ALP1
       IF(ARG.GT.-200.) S1=EXP(ARG)
       THET(KK)=EL
       K1=KK+1
       K2=KK-1
C
C COMPUTE TARGET VELOCITY AND ESTIMATION ERROR
C
       X3(K1)=X3(KK)+EDD*DEL
       IF(ALP1.EQ.0.) GO TO 4
       X4(K1)=S1*X4(KK)+EDD*(1.-S1)/ALP1
       GO TO 3
4      X4(K1)=X4(KK)+EDD*DEL
3      CONTINUE
       EDH(KK)=X3(KK)-X4(KK)
       IF(KK.GE.2) EDDH(KK)=(EDH(KK)-EDH(K2))/DEL
       X30(1)=X3(K1)
       COR=CO(1)*ALP2
       CR4=CO(1)*ALP4
       IF((KK-1).LE.M) GO TO 150
       AL1=1.+0.001*(5.2*TAU+0.486*TAU**2)*SIN(THET(KK-M))
       A(1,1)=-COR
       A(1,2)=-CO(1)*ALP3
       A(2,ND1)=A(1,1)*AL1*A2
       A(2,ND)=A(1,2)*AL1*A2
       DO 2 I=1,ND2
       J1=I+2
       A(J1,I)=RA
       A(J1,J1)=-RA
2      CONTINUE
       CALL DSCRT(ND,B,DEL,EB,EBINT,5)
       CR3=CR4*AL1*A2
       SCAL1=SCAL*(AL1*A2)**2
C
C COMPUTE MEAN TRACKING ERROR (I.E. LAG ANGLE)
C
       U=ALP2*W(1)+ALP3*W(2)+ALP4*(X3(KK)-X4(KK))
       DO 110 I=1,ND
       DO 120 J=1,ND
       D(I)=D(I)+EA(I,J)*W(J)
120    CONTINUE
110    CONTINUE
       F(1)=(1.-CR4)*X3(KK)+CR4*X4(KK)
       F(2)=X3(KK)+CR3*(X4(KK-M)-X3(KK-M))+0.001*(1.-A2)*(5.2+0.972*TAU
      1)*COS(THET(KK-M))
       DO 130 I=1,ND
       DO 140 J=1,2
       D(I)=D(I)+EAINT(I,J)*F(J)
140    CONTINUE
       W(I)=D(I)
       D(I)=0.
130    CONTINUE
C
C COMPUTE COVARIANCE MATRIX
C
       CQC(1,1)=(ALP5+ALP6*ABS(EDH(KK))+ALP7*ABS(EDDH(KK)))
      1 *SCAL
       CQC(2,2)=(ALP5D(KK-M)+ALP5*ABS(EDH(KK-M))+ALP7*ABS(EDDH(KK-M)
      1))*SCAL1
```

```
      CALL MULT(EAINT,CQC,ND,N1,P1,10)
      CALL MULT(EA,P,ND,N1,P2,13)
      DO 220 I=1,ND
      DO 220 J=1,ND
      P(I,J)=P1(I,J)+P2(I,J)
220   CONTINUE
150   CONTINUE
      ELERR=W(1)
      ELSD=SQRT(P(1,1))
      ELTR=W(2)
      RETURN
      END
      SUBROUTINE OBSA76(AZERR,AZSD,ELG)
      COMMON/S/C0(2),DEL,KK,ND,Y10(2),X30(2),EL,ELDD,AZDD,M,RA,
     1 A2,ND1,ND2,NA4,UEL,U,ELTR,AZTR,ISET,Z1,Z2,TAU,TS(15),TE(15),T,IBL
      DIMENSION W(4),P(4,4),P1(4,4),P2(4,4),ALPHA(7),A(4,4)
     1,B(16),EB(16),EBINT(16),EA(4,4),EAINT(4,4),F(4),CQC(4,4),D(4)
     2,X3(1000),X4(1000),EDH(1000),EDDH(1000),ALP50(1000)
      EQUIVALENCE (A(1,1),B(1)),(EA(1,1),EB(1)),(EAINT(1,1),EBINT(1))
      DATA WT/1./
      DATA ALPHA/5.5394,.13894,.17773,1.0353,.25286E-5,.19766E-3,.75785E
     A-3/
      IF((KK-1).GE.M) GO TO 6
      IF(KK.GT.1) GO TO 5
      N1=ND**2
C
C INITIALIZATION
C
      DO 1 I=1,ND
      DO 1 J=1,ND
      P(I,J)=0.
      CQC(I,J)=0.
1     A(I,J)=0.
      DO 11 I=1,ND
      W(I)=0.
      D(I)=0.
      F(I)=0.
11    P(I,I)=0.0000
      P(1,1)=(0.0059083*COS(ELG))**2
      P(2,2)=(0.0071597*COS(ELG))**2
      X3(1)=X30(2)
      X4(1)=0.
      EDH(1)=X3(1)
      EDDH(1)=0.
      S1=0.
      ALP1=ALP10=ALPHA(1)
      ALP2=ALPHA(2)
      ALP3=ALPHA(3)
      ALP4=ALPHA(4)
      ALP5=ALPHA(5)
      ALP6=ALPHA(6)
      ALP7=ALPHA(7)
      ALP50(1)=ALP5
      ACC=0.
      IFLAG=0
C
C COMPUTE AND STORE STATES X3 AND X4
C
5     W(1)=Y10(2)
6     IF(ISET.EQ.1) W(2)=Z2*COS(ELG)
      CONTINUE
      C9=COS(ELG)
      IF(IBL.LT.1) GO TO 18
      ALP5=ALPHA(5)
      IS=IFLAG+1
      DO 12 I=IS,IBL
```

```
      IF(T.GE.TS(I).AND.T.LT.TE(I)) GO TO 15
      ATT=AMIN1(1.5,ACC/3.)
      IF(T.GE.TE(I).AND.T.LT.(TE(I)+ATT)) GO TO 16
      IF(T.GE.(TE(I)+ATT)) GO TO 21
      ACC=0.
      IF(T.LT.TS(IS)) GO TO 18
      GO TO 12
16    ALP1=ALP10+(ALPHA(1)-ALP10)*(1.-EXP(-0.43*(T-TE(I))))
      ALP5=-0.001*(1.-EXP(-0.43*(T-TE(I))))
      GO TO 18
15    ACC=ACC+DEL
      IFLAG=I-1
      ALP1=ALPHA(1)*EXP(-1.0756*(T-TS(I)))
      ALP2=ALPHA(2)*EXP(-3.12*(T-TS(I)))
      ALP3=ALPHA(3)*EXP(-0.52*(T-TS(I)))
      ALP5=0.001*(1.-EXP(-0.12*(T-TS(I))))
      ALP10=ALP1
      GO TO 18
21    ALP1=ALPHA(1)
      ALP2=ALPHA(2)
      ALP3=ALPHA(3)
      ALP5=ALPHA(5)
      GO TO 18
12    CONTINUE
18    CONTINUE
      ALP5D(KK)=ALP5
      ARG=-DEL*ALP1*C3
      IF(ARG.GT.-200.) S1=EXP(ARG)
      K1=KK+1
      K2=KK-1
      X3(K1)=X3(KK)+AZDD*DEL
      IF(ALP1.EQ.0.) GO TO 4
      X4(K1)=S1*(X4(KK)+AZDD*DEL)
      GO TO 3
4     X4(K1)=X4(KK)+AZDD*DEL
3     CONTINUE
C
C COMPUTE AND STORE ESTIMATED TARGET VELOCITY AND ACCELERATION
C
      EDH(K1)=X3(K1)-X4(K1)
      IF(K2.GE.1) EDDH(KK)=(EDH(KK)-EDH(K2))/DEL
      X30(2)=X3(K1)
      IF((KK-1).LE.M) GO TO 130
      DO 2 I=1,ND2
      J1=I+2
      A(J1,I)=RA
      A(J1,J1)=-RA
2     CONTINUE
      THEBD=(ELG-DELG)/DEL
      CB=COS(ELG)
      TB=-THEBD*TAN(ELG)
      SCAL=(C0(2)*CB)**2/DEL
      COR=C0(2)*ALP2*CB
      A(1,1)=-COR*TB
      A(1,2)=-C0(2)*ALP3*CB
      A(2,ND1)=-COR*A2
      A(2,ND)=A(1,2)*A2
      A(2,2)=TB
C
C COMPUTE TRANSITION MATRIX EA AND ITS INTEGRAL EAINT
C
      CALL DSCRT(ND,3,DEL,EB,EBINT,5)
      CR4=C0(2)*ALP4*CB
C
C COMPUTE MEAN TRACKING ERROR
C
```

```
      U=ALP2*W(1)+ALP3*W(2)+ALP4*(X3(KK)-X4(KK))
      DO 110 I=1,ND
      DO 120 J=1,ND
      D(I)=D(I)+EA(I,J)*W(J)
120   CONTINUE
110   CONTINUE
      F(1)=(CB-CR4)*X3(KK)+CR4*X4(KK)
      F(2)=X3(KK)*CB+CR4*(X4(KK-M)-X3(KK-M))*A2
      DO 130 I=1,ND
      DO 140 J=1,2
      D(I)=D(I)+EAINT(I,J)*F(J)
140   CONTINUE
      W(I)=D(I)
      D(I)=0.
130   CONTINUE
C
C COMPUTE COVARIANCE MATRIX
C
      COC(1,1)=(ALP5+ALP6*ABS(EDH(KK))+ALP7*ABS(EDDH(KK)))
     1 *SCAL
      COC(2,2)=(ALP5)(KK-M)+ALP6*ABS(EDH(KK-M))
     1  +ALP7*ABS(EDDH(KK-M)))*SCAL*A2**2
      CALL MULT(EAINT,COC,ND,N1,P1,10)
      CALL MULT(EA,P,ND,N1,P2,1)
      DO 220 I=1,ND
      DO 220 J=1,ND
      P(I,J)=P1(I,J)+P2(I,J)
220   CONTINUE
150   CONTINUE
      OELG=ELG
      AZERR=W(1)/CB
      AZSD=SQRT(P(1,1))/CB
      AZTR=W(2)/CB
      RETURN
      END
      SUBROUTINE MULT(E,F,L,L1,H,MR)
      DIMENSION E(1),F(1),G(16),H(1)
      DO 10 I=1,L
      II=1
      DO 10 K=1,L
      TEMP=0.
      DO 5 J=I,L1,L
      TEMP=TEMP+E(J)*F(II)
5     II=II+1
      KK=(K-1)*L+I
      H(KK)=TEMP
10    G(KK)=TEMP
      IF(MR.EQ.1)RETURN
      DO 20 I=1,L
      DO 20 K=I,L
      TEMP=0.
      II=K
      DO 15 J=I,L1,L
      TEMP=TEMP+G(J)*E(II)
15    II=II+L
      KK=(K-1)*L+I
20    H(KK)=TEMP
      L2=L-1
      DO 30 I=1,L2
      L3=I+1
      DO 30 J=L3,L
      K1=(I-1)*L+J
      K2=(J-1)*L+I
30    H(K1)=H(K2)
      END
      SUBROUTINE DSCRT(NDIM,A,DEL,EA,EAINT,NT)
```

```
      DIMENSION A(1),EA(1),EAINT(1),COEF(30)
C        SETS EA=EXP(A*DEL),EAINT=INTEGRAL EA 0 TO DEL
      NDIM1=NDIM+1
      NN=NDIM*NDIM
      NTM1=NT-1
      COEF(NT)=1.
      DO 10 I=1,NTM1
      II=NT-I
   10 COEF(II)=DEL*COEF(II+1)/FLOAT(I)
C           NT MUST BE AT LEAST 3
      CALL DIAG(NDIM,EAINT,A,COEF(1),COEF(2))
      DO 60 L=3,NT
      CALL MULT(A,EAINT,NDIM,NN,EA,1)
      IF(L.EQ.NT)GO TO 70
   60 CALL DIAG(NDIM,EAINT,EA,1.0,COEF(L))
   70 DO 80 II=1,NN,NDIM1
      EA(II)=EA(II)+1.0
   80 CONTINUE
      END
      SUBROUTINE DIAG(NDIM,A,B,C1,C2)
      DIMENSION A(1),B(1)
      NDIM1=NDIM+1
      NN=NDIM*NDIM
      NM1=NDIM-1
      II=1
      IF(C1.EQ.1.0) GO TO 10
      DO 5 J=1,NN,NDIM
      K=J+NM1
      DO 4 I=J,K
    4 A(I)=C1*B(I)
      A(II)=A(II)+C2
    5 II=II+NDIM1
      RETURN
   10 DO 7 J=1,NN,NDIM
      K=J+NM1
      DO 6 I=J,K
    6 A(I)=B(I)
      A(II)=A(II)+C2
    7 II=II+NDIM1
      RETURN
      END
600,2.46,2,1
8.01,15.51
```

REFERENCES

Banks, H. T. and J. A. Burns, March 1978, "Hereditary Control Problems: Numerical Methods Based on Average Approximations," SIAM J. Control and Optimization, Vol. 16, No. 2, pp. 169-208.

Kleinman, D. L. and T. R. Perkins, August 1974, "Modeling Human Performance in a Time-Varying Anti-Aircraft Tracking Loop," IEEE Trans. Automat. Contr., Vol. AC-19, No. 4, pp. 297-306.

Kou, R. S., B. C. Glass, M. M. Vikmanis, M. S. Moran, and C. N. Day, 1978, "A Linear Time-Varying Antiaircraft Gunner Model Based on Observer Theory," Proceedings of the International Conference on Cybernetics and Society, Vol. II, pp. 1421-1426.

Luenberger, D. G., December 1971, "An Introduction to Observers," IEEE Trans. Automat. Contr., Vol. AC-16, No. 6, pp. 596-602.

Phatak, A., H. Weinert, I. Segall, and C. N. Day, 1976, "Identification of a Modified Optimal Control Model for the Human Operator," Automatica, Vol. 12, pp. 31-41.

Wei, K. C., March 1981, A Human Operator Gunner Model for Tracer-Directed Antiaircraft Artillery Fire, AFAMRL-TR-80-142, Air Force Aerospace Medical Research Laboratory, Wright-Patterson AFB, Ohio. AD A097 824.

Yu, C. F., K. C. Wei, and M. M. Vikmanis, May 1980, "A Gunner Model for an AAA Tracking Task with Interrupted Observations," Proceedings of the 16th Annual Conference on Manual Control, Cambridge, Massachusetts.

Yu, C. F., A Human Gunner Model for a Tracking Task with Interrupted Obserations in an Antiaircraft Artillery System, AFAMRL-TR-81-37, Air Force Aerospace Medical Research Laboratory, Wright-Patterson AFB, Ohio. AD A097 360.

U.S. GOVERNMENT PRINTING OFFICE: 1981—757-002/26

DATE
ILMED
-8